Anita Sellent

Dense Correspondence Field Estimation from Multiple Images

Dissertation

Dense Correspondence Field Estimation from Multiple Images

Von der Carl-Friedrich-Gauß Fakultät
Technische Universität Carola-Wilhelmina zu Braunschweig

zur Erlangung des Grades

Doktor Ingenieur (Dr.-Ing.)

genehmigte

Dissertation

von	Anita Sellent
geboren am	1. Oktober 1980
in	Kirchen

Eingereicht am: 28. Januar 2011

Mündliche Prüfung am: 15. April 2011

Referent: Prof. Dr.-Ing. Marcus Magnor

Koreferent: Prof. Dr.-Ing. Bodo Rosenhahn

(2011)

Anita Sellent: Dense Correspondence Estimation from Multiple Images

Die Edition Octopus erscheint im Verlagshaus
Monsenstein und Vannerdat OHG Münster
www.edition-octopus.de

Satz und Umschlag: A. Sellent
Illustrationen: A. Sellent
Druck und Bindung: MV-Verlag

ISBN

Abstract

Most optical flow algorithms assume pairs of images that are acquired with an ideal, short exposure time. We present two approaches, that use additional images of a scene to estimate highly accurate, dense correspondence fields. In our first approach we consider video sequences that are acquired with alternating exposure times so that a short-exposure image is followed by a long-exposure image that exhibits motion-blur. With the help of the two enframing short-exposure images, we can decipher not only the motion information encoded in the long-exposure image, but also estimate occlusion timings, which are a basis for artifact-free frame interpolation. In our second approach we consider the data modality of multi-view video sequences, as it commonly occurs, e.g., in stereoscopic video. As several images capture nearly the same data of a scene, this redundancy can be used to establish more robust and consistent correspondence fields than the consideration of two images permits.

Kurzfassung

Die meisten Verfahren zur Schätzung des optischen Flusses verwenden zwei Bilder, die mit einer optimalen, kurzen Belichtungszeit aufgenommen wurden. Wir präsentieren zwei Methoden, die zusätzliche Bilder zur Schätzung von hochgenauen, dichten Korrespondenzfeldern verwenden. Die erste Methode betrachtet Videosequenzen, die mit alternierender Belichtungsdauer aufgenommen werden, so dass auf eine Kurzzeitbelichtung eine Langzeitbelichtung folgt, die Bewegungsunschärfe enthält. Mit der Hilfe von zwei benachbarten Kurzzeitbelichtungen können wir nicht nur die Bewegung schätzen, die in der Bewegungsunschärfe der Langzeitbelichtung verschlüsselt ist, sondern zusätzlich auch Verdeckungszeiten schätzen, die sich bei der Interpolation von Zwischenbildern als große Hilfe erweisen. Die zweite Methode betrachtet Videos, die eine Szene aus mehreren Ansichten aufzeichnen, wie z.B. Stereovideos. Dabei enthalten mehrere Bilder fast dieselbe Information über die Szene. Wir nutzen diese Redundanz aus, um konsistentere und robustere Bewegungsfelder zu bestimmen, als es mit zwei Bildern möglich ist.

Summary

Many applications in computer vision and video post-production require dense correspondence fields between images of a video stream. Most state-of-the-art algorithms estimate these correspondences by assuming pairs of images that are acquired with an idealized, infinitively short exposure time. In our work we present two approaches that use additional images of a scene to estimate highly accurate, dense correspondence fields.

In our first approach we consider video sequences that are acquired with alternating exposure times so that a short-exposure image is followed by a long-exposure image that exhibits motion-blur. With the help of the two enframing short-exposure images, we can decipher not only the motion information encoded in the long-exposure image, but also estimate occlusion timings, which are a prerequesite for artifact-free frame interpolation. We develop a suitable image formation model which relates the long-exposure image to preceding and succeeding short-exposure images in terms of dense pixel correspondences and per-pixel occlusion/disocclusion timings. Based on this image formation model, we describe and compare two algorithms to estimate the motion field not only for visible image regions but also for regions that become occluded.

In our second approach we consider the data modality of multi-view video sequences, as it commonly occurs, e.g., in stereoscopic video. If

several images capture nearly the same data of a scene, this redundancy can be used to establish more robust and consistent correspondence fields than the consideration of two images permits. We use the redundancy in the image data to establish a confidence measure based on the consistency of the correspondences on a loop of three images. While usually confidence measures are applied to sparsify correspondence fields after the estimation is completed, we include our confidence measure directly into the estimation process and obtain dense correspondence fields with increased accuracy. Furthermore, the same concept of consistency can be employed to establish robust feature matches for applications in which epipolar geometry cannot be used to eliminate outliers. Finally, we show that the condition of consistency provides a constraint that is sufficiently strong to directly combine the strength of sparse feature matching and dense correspondence estimation in a common framework without further consideration of outliers.

Zusammenfassung

Viele Anwendungen in der Computer Vision und der Nachbearbeitung von Videos benötigen dichte Korrespondenzen zwischen den Bildern eines Videostroms. Nach dem Stand der Technik werden diese Korrespondenzen mit Hilfe von Bildpaaren bestimmt, die mit einer optimal kurzen Belichtungszeit aufgenommen sind. In unserer Arbeit stellen wir zwei Ansätze vor, die zusätzliche Bilder einer Szene verwenden, um hochgenaue Bewegungsfelder zu bestimmen.

In unserem ersten Ansatz betrachten wir Videosequenzen, die mit alternierender Belichtungszeit aufgenommen werden, so dass einer Kurzzeitbelichtung eine Langzeitbelichtung mit Bewegungunschärfe folgt. Mit der Hilfe von zwei benachbarten Kurzzeitbelichtungen können wie nicht nur die Bewegung schätzen, die in der Bewegungsunschärfe der Langzeitbelichtung verschlüsselt ist, sondern zusätzlich auch Verdeckungszeiten, die eine Grundvoraussetzung für artefaktfreie Bildinterpolation sind. Wir entwickeln ein geeignetes Bildentstehungsmodell, das die Langzeitbelichtung, die vorangehende Kurzzeitbelichtung und die nachfolgende Kurzzeitbelichtung über ein dichtes Bewegungsfeld und pixelweise Verdeckungszeiten zueinander in Beziehung setzt. Darauf basierend beschreiben und vergleichen wir zwei Algorithmen, die das Bewegungsfeld nicht nur für die sichtbaren Bildregionen schätzt, sondern auch für Bildregionen, die in einer der Kurzzeitbelichtungen verdeckt sind.

In unserem zweiten Ansatz betrachten wir die Datenlage für eine Szene, die mit mehreren Videokameras aufgezeichnet wird. Dies ist z.B.

bei Stereovideos der Fall. Da mehrere Bilder fast dieselben Daten der Szene aufzeichnen, können wir diese Redundanz benutzen, um konsistentere und robustere Bewegungsfeld zu bestimmen, als es die Betrachtung von nur jeweils zwei Bildern zuläßt. Wir nutzen die Redundanz der Bilddaten aus, um ein Zuverlässigkeitsmaß einzuführen, das auf der Konsistenz der Korrespondenzen auf einer Schleife von drei Bildern basiert. Normalerweise werden Zuverlässigkeitsmaße nach Beendigung der Korrespondenzschätzung ausgewertet, um unzuverlässige Korrespondenzen zu eliminieren. In unserem Ansatz hingegen wird das Zuverlässigkeitsmaß während der Berechnung der Korrespondenzen ausgewertet, und wir erhalten dichte Korrespondenzfelder mit erhöhter Genauigkeit. Dasselbe Konzept der Konsistenz kann auch für das Verknüpfen von Bildmerkmalen verwendet werden, insbesondere in Situationen, die die Anwendung der Epipolargeometrie zur Verwerfung von falschen Zuordnungen nicht zulassen. Abschliessend können wir zeigen, dass die Bedingung der Konsistenz so stark ist, dass das Verknüpfen von Bildmerkmalen und dichte Korrspondenzschätzung in einem gemeinsamen Algorithmus vereinbar sind, der keine weiteren Betrachtungen von falschen Zuordnungen benötigt.

Acknowledgements

Diese Arbeit wurde von vielen Menschen auf sehr unterschiedliche Weise unterstützt. Zunächst natürlich von Marcus, der mich, immer wenn ich wirklich ratlos zu ihm gekommen bin, mit neuen Ideen davonstürmen sah und mir auch bei vielen kleineren Fragen stets mit gutem Rat zur Seite stand. Eine sehr wertvolle Unterstützung waren auch die Coautoren, mit denen ich an Veröffentlichungen gearbeitet habe und die mit mir die Wege zu neuen Ideen beschritten haben. Zusammen die Kreuzungen auf dem Weg zu entdecken und zu diskutieren, welcher Weg der vielversprechendste ist, hat mir nicht nur geholfen, mich für einen Weg zu entscheiden, sondern auch die Landschaft, durch die die Wege führen, besser zu verstehen.

Unterstützt haben diese Arbeit alle meine Kollegen - zum einen durch die Versorgung mit Muffins und dringend notwendigen Tee- und Kaffeepausen, zum anderen auch durch die Weitergabe ihrer Erfahrungen und durch den gemeinsamen Austausch. Besonderer Dank gilt Christian Lipski, der die Testdatensätzen zu diversen Problemen hergestellt hat, und meinen Korreskturlesern Tobi, Sven, Martin und Christian Linz, die mir mit ihrer konstruktiven Kritik sehr geholfen haben.

Finanziert wurde diese Arbeit von der Deutschen Forschunggemeinschaft, vorangetrieben von all jenen Menschen um mich herum, die mit mir zusammen das eine oder andere Feierabendbier genossen haben. Für besondere Unterstützung möchte ich mich bei meinen Eltern bedanken, die mir die Freiheit schenken, das zu machen, was mir Spaß macht.

Contents

Chapter 1

Introduction

1.1 Motivation

In our work we consider dense correspondence estimation from images. But why do we actually need to estimate correspondences from images? Why do we need them to be dense, i.e., defined for every single pixel? And what can we do with the estimated correspondence fields? We are going to give an introduction into this topic in the following section.

The need for dense correspondence fields

In recent years, digital video has become nearly ubiquitous [CGG+06]. With the availability of large quantities of storage space, fast data transmission and cheap consumer video cameras, digital video has gained a large degree of popularity. For example, digital video cameras are integrated in nearly every cell-phone today.

For the human observer, the temporal dimension of a video sequence is immediately accessible. Played back at a sufficiently high framerate, humans perceive smooth motion in a sequence of images [MF88]. From the temporal dimension of the sequence, humans can easily derive additional information about the scene such as relative motion of objects, the structure of objects and their reflectance properties [Ull79].

Technically, a video sequence is recorded as a sequence of still images. Pixels are read out from the sensor independently for each frame and treated as disjoint measurements. However, starting early in the video processing pipeline 2D correspondences between successive images are valuable information: Compression algorithms store a video sequence not as a sequence of independent images, but use correspondences to exploit the temporal coherence in a video sequence [Ric03]. As these algorithms mainly need to be fast and perform computations on the level of pixels values, they usually settle for rough motion estimation [Ric03].

In contrast, more complex algorithms such as camera-based driver assistant systems [GCTC98, KZK03] or video editing tools [Ado10, Bor10] need pixel correspondences that accurately identify the projections of a moving 3D scene point over time. If the correspondences for every pixel in the image are known, i.e. the correspondence field is dense, arbitrarily shaped objects can be reliably identified as obstacles or edited consistently over a video sequence, Fig 1.1. The output of these image processing systems depend highly on the quality of the pixel correspondence estimation. Every time correspondence estimation fails, e.g., due to rapid motion, changing lighting conditions or temporal occlusion, reinitialization is necessary. But not only complete failure in correspondence estimation requires reinitialization. Also, inaccuracies of pixel correspondences build up over time, leading to drift in the scene points [RAKRF08]. Thus, the dense correspondence field is required to be not only robust but also highly accurate.

Once highly accurate, dense correspondences are available, the applications are numerous. In superresolution algorithms, dense, highly accurate correspondences are required to identify the information collected in multiple images [IP91]. For video stabilization, dense correspondences are required over an entire video sequence to render stabilized frames faithfully [MOG+06]. Structure-from-motion approaches use dense correspondences to obtain information on the structure of the scene [XCK06]. Similarly, dense correspondences are used in robot navigation to avoid obstacles and for visual odometry [DK02, MB04]. Combining dense 2D correspondences from several cameras also allows to estimate the 3D motion of

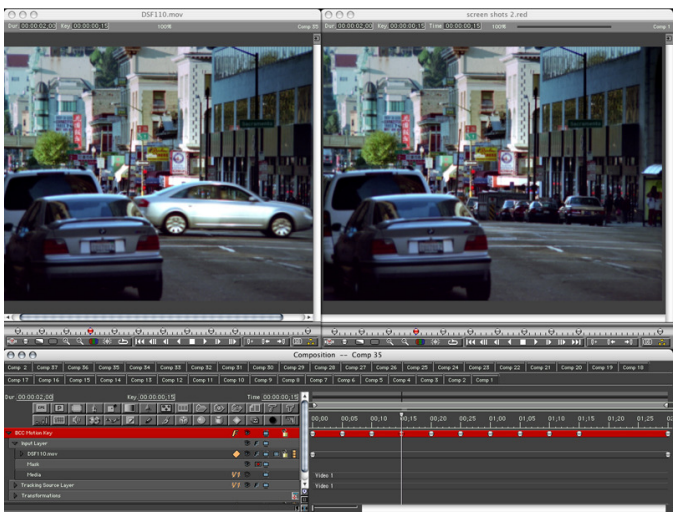
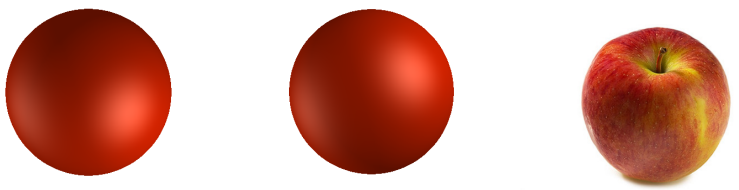


Figure 1.1: Professional video editing systems such as Red from Boris FX [Bor10] use dense pixel correspondences to propagate editing operations from one keyframe automatically to all succeeding and/ or preceding frames in a video sequence.

the scene [VBR+05] which can be used, e.g., in textile reconstruction [SM04] or obstacle recognition [WRV+08]. In augmented-reality scenarios, dense pixel correspondences allow to track natural image features so that no artificial markers need to be introduced into the scene [NYC+99, MYN07].

Problems in dense correspondence estimation

Since dense correspondences between images are useful and needed in many domains, their estimation is still an active field of research. Actually, the problem of estimating motion based on images contains ambiguities, which even human observers - with all their knowledge of the real world - cannot always resolve. For example, consider the image of a uniformly colored sphere in Fig. 1.2(a). A video of this sphere rotating in any direction around its center results in a sequence of identical images. They cannot provide any clue whether the sphere is rotating



(a) (b) (c)

Figure 1.2: (a) For a rotating, uniformly colored sphere, motion is not apparent and all images of a video sequence look alike. (b) Moving a light source around an otherwise static scene causes apparent motion. (c) Luckily, most natural objects provide enough texture so that apparent motion can serve as an approximation of the actual 2D motion.

or static; the motion is visually imperceptible. Reversely, there is a whole set of motions, e.g., no motion, clockwise rotation, anti-clockwise rotation etc. that are able to produce a sequence of identically looking images.

Luckily, truly uniform surfaces are rare in natural scenes, Fig. 1.2(c), and object texture helps to solve the ambiguity of the motion. But objects might be only partially textured and have uniform color in other regions. An intuitive way to provide motion information to untextured regions is to interpolate between correspondences found in textured regions. However, this approach does not need to be correct, as usually an image depicts several objects which move independently in a scene and thus have different 2D correspondences. So correspondence interpolation algorithms have to be carefully designed not to interpolate between different objects. Of course, if these objects are visually indistinguishable, boundaries between these objects cannot be easily detected.

Another problem in estimating correspondences between two images is the actual description of the entities that should correspond. The most intuitive and fine-grained description is the color or gray-value of individual pixels. Although promising high resolution of the correspondence field, single pixel color is very susceptible to noise, changes in illumination and moving shadows. In Fig. 1.2(a)

shading effects cause the image of the 3D sphere to have some variation in the color, although the whole sphere has uniform albedo. If in this setup the light source is moved, Fig. 1.2(b), color values change and motion of the sphere is perceived even though the sphere itself may remain static. Other descriptions based on larger image regions and the gradient of gray-values rather than the gray-values themselves, e.g., SIFT [Low99] or SURF [BETV08] features, provide robustness to changes in illumination and noise. Yet, the larger the image area needed to describe a feature, the less stable it is with respect to the detection of motion boundaries and to partly occluded objects. Pixels that influence the description of a region might in fact belong to a different object moving in a different direction, or they might not be visible in the next frame.

Generally, occlusions that can occur due to different objects moving in different directions, different parts of the same objects occluding each other or objects leaving the field of view of the camera, create a problem for correspondence estimation. Points that become occluded do not have any correspondence in the next frame. Nevertheless, motion information is needed also for these points, e.g., for frame interpolation. Furthermore, enforcing correspondences for occluded points might drive the motion estimation algorithm away from the desired solution.

All the aforementioned considerations implicitly assume the minimal example of two frames to establish correspondences between them. The question arises, *how can information, captured by additional images help to establish more robust correspondences? Can additional images be used to gain more information about the scene - information that can be useful to improve, for example, frame interpolation?* Before we consider these questions in more detail, we first give an overview over dense motion estimation and review the state-of-the-art in this field.

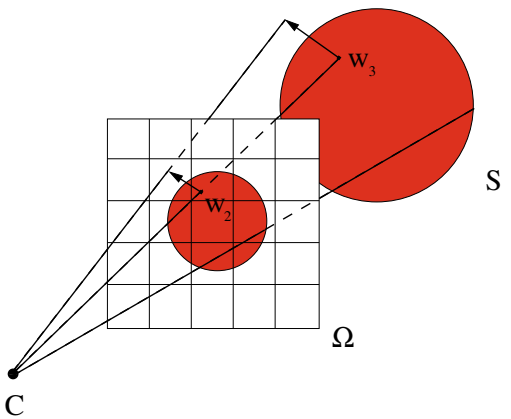


Figure 1.3: The projection of the 3-dimensional scene S onto the image plane Ω also projects the 3-dimensional scene motion w_3 to the 2-dimensional motion w_2 in the image plane.

1.2 Background

When objects in our 3D world move, the difference between their positions at two instances t and $t + \Delta t$ in time can be described by a dense field of 3D vectors. Every one of these vectors approximates the displacement of a 3D world point over the time interval Δt. In the following, we restrict the dense field of 3D motion vectors to the visible scene surfaces to obtain a sparser representation.

An image is a projection of the visible surfaces in the 3D world to the 2D image plane. Accordingly, the 3D motion vectors on object surfaces are projected to 2D motion vectors, Fig. 1.3. However, more than one 3D surface point may be projected to the same pixel in the image, so the 2D motion vector may not be well-defined for all image points. For example, for transparent objects, a point in the image receives the projection of the transparent object as well as the projection of the object visible behind the transparent object. The 2D motion field at this location should therefore account for both, the motion of the transparent and the motion of the background object. To restrict the solution space of the

motion estimation problem, we here assume that all objects are opaque so that the projection of the 3D motion field is indeed a 2D motion field that is well-defined for every point in the image. Still, this 2D motion field is undiscernible from images alone as different 2D motion fields may produce the same sequence of images, Fig. 1.2. All that can be observed in images are the changes in the brightness pattern. This apparent motion is known as *optical flow*. As most natural objects provide at least some texture, i.e., patterns in the brightness distribution that are independent of the current position of the object, optical flow is frequently used as a fair approximation of the actual 2D motion field.

Choosing the brightness of a pixel as our description of the image, we denote the temporally varying brightness pattern on the image sensor with $I : \Omega \times \mathbb{R} \to \mathbb{R}$ where $\Omega \subset \mathbb{R}^2$ is the image domain. The assumption that a scene point maintains its description, i.e., its brightness, in the image although it changes its location can then be expressed in terms of its apparent motion $(u, v) \in \mathbb{R}^2$ as

$$I(x, y, t) = I(x + u, y + v, t + \Delta t) \ . \tag{1.1}$$

Note that, on the one hand, different pairs of real values (u, v) may satisfy this equation, and that, on the other hand, considering shading of 3D objects, the projection of the 3D motion field to the image plane might actually not be one of these solutions. Yet, in a wide range of scenarios the brightness constancy assumption is satisfied by a large number of scene points, so that the optical flow represents a reasonable approximation to the 2D motion field that is generated by the actual 3D scene motion.

Applying a Taylor series expansion to the image function $I(x+u, y+v, t+\Delta t)$ and ignoring the dependency of the optical flow on time and location results in

$$I(x+u, y+v, t+\Delta t) = I(x, y, t) + \frac{\partial I(x, y, t)}{\partial x} u + \frac{\partial I(x, y, t)}{\partial y} v + \frac{\partial I(x, y, t)}{\partial t} \Delta t + \ldots \tag{1.2}$$

If the higher order terms in the Taylor expansion are dismissed and the brightness constancy assumption holds we obtain

$$\frac{\partial I(x,y,t)}{\partial x}u + \frac{\partial I(x,y,t)}{\partial y}v + \frac{\partial I(x,y,t)}{\partial t}\Delta t \approx 0 \tag{1.3}$$

which is the basic data-term for optical flow estimation. Denoting the spatial gradient of the brightness function with

$$\nabla_2 I = \begin{pmatrix} \frac{\partial I(x,y,t)}{\partial x} \\ \frac{\partial I(x,y,t)}{\partial y} \end{pmatrix} \tag{1.4}$$

the optical flow equation can be written as

$$\nabla_2 I^T \cdot \begin{pmatrix} u \\ v \end{pmatrix} + \frac{\partial I(x,y,t)}{\partial t}\Delta t \approx 0 \ . \tag{1.5}$$

If the spatial image gradient is different from zero this equation imposes one constraint for the two unknowns of the motion vector (u, v). Looking at Eq. (1.5) we see that actually only the motion component in the direction of the spatial gradient of the brightness can be estimated. The motion component perpendicular to the spatial gradient cannot be estimated. This phenomenon is known as the *aperture problem* [VP89]. Different authors propose different additional assumptions to solve for both components of the motion vectors, Sect. 1.3, usually by imposing additional constraints on the spatial gradient of the optical flow.

Apart from the approximate quality of optical flow and the aperture problem, there are further problems bothering practical 2D correspondence estimation. First, the formulation of the brightness constancy assumption doesn't allow to consider occlusion or disocclusion which occur if parts of the scene move with different motions relative to the camera. Second, the suppression of higher order terms of the Taylor expansion in the derivation of Eq. (1.3) assumes the motion to be small. In fact, the consideration of the temporal derivative requires the motion to be smaller than one pixel per frame, as otherwise temporal aliasing is introduced, Sect. 2.3. In this thesis we are going to look at how these problems can be addressed.

1.3 State of the Art

Optical flow computation has a long standing history in computer vision research. A huge number of papers on optical flow estimation has been published during the last 30 years. Various surveys on optical flow research covering the state-of-the-art of the time can be found [AN88, ON94, SK99]. Standard test sets [BFB94, BSL+07] allow to evaluate the algorithms quantitatively and compare their performance [MB96, MNCG01, BSL+09].

In fact, starting with the seminal work by Horn and Schunck [HS81] and Lucas and Kanade [LK81] much attention has been devoted to different aspects of optical flow estimation. The approach of Horn and Schunck as well as that of Lucas and Kanade are based on the brightness constancy assumption in Eq. (1.3), but they differ in their treatment of the aperture problem. While Horn and Schunck [HS81] propose to regularize the motion field with the squared norm of its gradient, Lucas and Kanade [LK81] assume that all pixels (x, y) in a fixed-sized window move with the same flow and construct an over-determined system of equations. As neither the assumption of smoothness nor the assumption of local constancy hold at motion boundaries, they are today replaced by robust [BA96, MPI96, ZPB07] and anisotropic [NE86, WTP+09] regularization approaches. Notably, Irani could show that for rigid scene objects, motion fields between multiple frames reside in a low-dimensional subspace [Ira02]. In this case exploiting the information given by the additional images solves the aperture problem and renders regularization as an additional constraint superfluous.

Originally, also the data-term on the left-hand side of Eq. (1.3) was considered in the squared norm [HS81, LK81]. As this penalization is susceptible to failures in the image formation model, such as changes in the brightness of a scene point's projection and sensor noise [HRRS86], the squared norm is today mostly replaced by robust penalizer functions that are more tolerable towards outliers [BA96, MPI96, BBPW04, ZPB07]. Also, different ways to describe the

image elements that are to be matched have been proposed in the literature to provide robustness to different illuminations. For example, Mileva et al. [MBW07] use pixel-accurate photometric invariants in their optical flow approach while, e.g., Liu et al. [LYT+08] use the region-based SIFT descriptor, trading accuracy in the optical flow for robustness in the motion estimation.

Scale-space approaches [Ana89] and iterative warping [AWS00, BBPW04] allow optical flow approaches to cope with fast motions, provided that the object is sufficiently large. These approaches fail for small, fast moving objects. Eliminating high image frequencies and down-sampling successively, small objects become indiscernible from the background before a level of the image pyramid is reached that can cope with fast motion. To detect large motions also for small objects, Steinbrücker et al. [SPC09] and also Linz et al. [LLM10] conduct an expensive full search in the image space while Xu et al. consider additional SIFT matches each time the algorithm proceeds to a finer level, tediously reconsidering different, non-local initializations on each level [XJM10]. More efficiently, Brox et al. [BBM09, BM10] include feature matches as a prior for dense optical flow estimation, detecting more motion details, but also struggling with wrong matches that deteriorate the overall quality of the optical flow. Lim et al. [LAG05] use a high-speed camera to record a temporally oversampled video. This way, motion between individual high-speed frames is sufficiently small to allow direct application of Eq. (1.3) and motion between main frames can be obtained by concatenation. But in spite of global correction steps, errors in the individual flow fields render the concatenation very noisy.

Some motion estimation algorithms consider several frames. Temporal Kalman-filtering between succeeding frames of a video sequence can be applied to obtain an initial estimate for the motion field [GJ97]. Temporal coherence is also used as an additional regularization constraint for optical flow estimation [BA91, WS01]. But even though considerable improvements in the optical flow can be obtained, none of the approaches consider occlusion effects.

In contrast, some two-image optical flow algorithms model occlusion explicitly [XCS+06, ADPS07, ST08]. Xiao et al. [XCS+06] and Sand and Teller [ST08] consider the optimization residual and the divergence of the estimated flow to detect occluded points, while Alvarez et al. [ADPS07] introduce a new term into optical flow estimation by considering the symmetry of forward and backward flow. All of these approaches extrapolate motion estimates to occluded areas by spatial filtering. This is motivated by the problem that the two-image setup does not contain any information on the motion of occluded points. Motion has to be interpolated based on the assumption of similar motion of similarly colored pixels.

While many problems in optical flow estimation have been addressed, the results are still not sufficiently robust and accurate for many applications. In this work we will provide two approaches that tackle the following problems:

- Where do occluded points move to?
- How can we deal with large motion?
- Where is optical flow estimation reliable?

In the next section we give an overview over the contribution of our work. Details are elaborated in Ch. 2 and 3. In the corresponding chapters we also discuss further literature that is related to the respective ideas but not classically related to optical flow estimation.

1.4 Overview and Contributions

The key idea of our approaches to dense correspondence estimation is to use more than two images and provide additional and a different kind of information than available in the usual two-image approach. In computational photography the basic concept is to spend some of the resources available to gain a different kind of information about a scene [LSC+08]. In our case, the desired information is the

Figure 1.4: An often involuntary way to capture motion is in the form of motion-blur when an image is exposed for an extended period of time.

dense correspondence or motion field. Apart from comparing succeeding frames in a video sequence, an intuitive - and often involuntary - way to capture motion is to expose an image sensor for an extended period of time. Moving objects in these images appear blurred in the direction of their motion, Fig. 1.4. Acquiring a sequence of only long-exposure images would trade off high frequencies in the scene against additional information on the scene's motion. Instead, in our approach, temporal resolution of a video camera is traded off for the additional motion information by alternating between short- and long-exposure images. This way, both high frequencies and motion information are recorded. In addition to the motion information, long-exposure images capture occlusion information. From occlusion information, short-exposure frames for any point in time between the two recorded short-exposure images can be interpolated. This new approach to motion and occlusion estimation and its signal theoretic background are presented in Ch. 2.

In Ch. 3 we look more closely at the data gathered by stereoscopic and multi-view video cameras. Due to the omnipresence of video cameras, fast transmission and readily available storage space, multi-view video data is becoming a common occurrence today [CGG+06]. Stereo cameras are also required to record content for stereoscopic cinema and other modern 3D display devices. In the multi-view video setup correspondences are to be established not only between

succeeding frames of a single video sequence but also between different cameras. While the alternate exposure approach of Ch. 2 can provide correspondences in the temporal direction, intermediate images between different cameras cannot be recorded with traditional cameras. Yet, using several video cameras, there is an abundance of images that show the same scene. Conveniently, the solution space of the correspondence problem is restricted by mutually neighboring images: correspondences between one pair of images must be in accordance with correspondences to the neighboring images. The advantages of considering spatially and temporally neighboring images are two-fold. First, the redundancy can be used to estimate more robust flow fields. Second, occluded points do not satisfy any consistency constraint between the images. They can thus be detected and assigned motion estimates that are not based on the brightness constancy assumption that they cannot correctly satisfy but on the assumption of motion similarity of similarly colored pixels.

In this work, the question of how correspondence estimation can be improved with additional images is considered and evaluated. Part of this work has been presented at the *IEEE International Conference on Computational Photography 2009* [SEM09] the *Vision, Modeling and Visualization Workshop 2009* [SEG+09] and the *IEEE International Conference on Image Processing 2010* [SLM10] or is accepted for publication in the *IEEE Transactions on Pattern Analysis and Machine Intelligence* [SEG+ar] and the *Journal of WSCG* [SEM11]. These publications are the foundation of this thesis which incorporates them under the framework of multi-image correspondence field estimation and presents an in-depth analysis of the approaches, together with updated results and discussions.

Chapter 2

Motion Estimation from Alternate Exposure Imaging

2.1 Introduction

Estimating the dense motion field between two consecutive images has been a heavily investigated field of computer vision research for decades [AN88, ON94, BFB94, MB96, SK99, MNCG01, BSL+07, BSL+09]. To approximate the actual 2D motion field, typically the optical flow between consecutive video frames is estimated. As local derivatives in space need to be numerically evaluated to solve the optical flow equation, optical flow algorithms work best with pinpoint-sharp images as input, i.e., with images depicting a dynamic scene at two discrete points in time. If regarded individually, however, short-exposure images capture no motion information at all. Instead, traditional optical flow methods reconstruct motion indirectly by motion-modeling the image difference.

Sampling theoretic considerations show that this approach is prone to temporal aliasing if the maximum 2D displacement in the image plane exceeds one pixel, i.e., twice the spatial bandlimit [Chr00]. To prevent aliasing, multi-scale optical flow methods pre-filter the image globally in the image domain because the motion is a priori unknown [Chr00]. This, however, is not the correct temporal filter: high spatial frequencies should be suppressed only in those Fourier

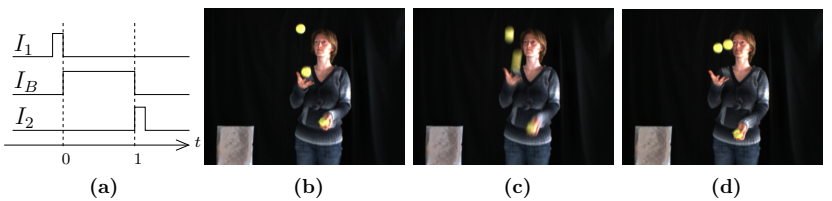


Figure 2.1: Alternate exposure imaging: (a) exposure timing diagram of (b) a short-exposure image I_1 followed by (c) a long-exposure image I_B and (d) another short-exposure image I_2.

domain regions where aliasing actually occurs, i.e., only in the direction of local motion.

There exists a simple way to achieve correct temporal pre-filtering by exposing the image sensor for an extended period of time. In long-exposure images high frequency components of moving objects are suppressed only in motion direction. Apart from circumventing the problem of temporal aliasing, long-exposure images bear the advantage that occlusion enters directly into the image formation process. A scene point and its motion contribute to a motion-blurred image exactly for as long as the point is not occluded. Only recently have optical flow algorithms begun to address occlusion [XCS+06, ST08, ADPS07], assigning occlusion labels per pixels. The moment of occlusion, however, cannot be easily determined from short-exposure images.

Inspired by these observations, we present an extension to traditional optical flow estimation. As input, our method requires images taken with different exposure times. An intermediate long-exposure image I_B is preceded and succeeded by two short-exposure images I_1 and I_2, Fig. 2.1. The intermediate motion-blurred image records the motion of the scene points directly, while the short-exposure images capture all high frequency details of the scene. In contrast to the severely underdetermined problems of motion estimation from two short-exposure images and image deblurring from motion-blurred images, motion estimation from alternate exposure imaging, although still underdetermined, has more scene in-

formation available. Taking advantage of the information contained in each of the image types, we introduce an image formation model that is equally valid for occluded and non-occluded points and allows for occlusion time estimation.

Before going into more detail, we first review previous work in motion estimation based on one or more motion-blurred images and discuss drawbacks and limitations of these approaches. The inherent limitations of both only motion-blurred and only sharp images for motion estimation become apparent when the space-time Fourier transform of a moving image sequence is considered. The basic analysis is presented in Sect. 2.3, before the alternate exposure image formation model is introduced in Sect. 2.4. For the numerical evaluation of the image formation model we propose two algorithms. In Sect. 2.5 we make some additional assumptions and derive a pointwise numerical algorithm for motion estimation. In Sect. 2.6 we alleviate the assumptions and present a regularized global minimization problem together with an elegant solution scheme. Both solution approaches are compared to each other and to the state-of-the art in motion detection in Sect. 2.7.

2.2 Related Work

The way two-image optical flow algorithms deal with the problems of occlusion and large motion was already discussed in Sect. 1.3. But motion estimation is also possible from a single, motion-blurred image. Assuming spatially invariant, constant velocity motion, Yitzhaky and Kopeika determine direction and extend of motion-blur via autocorrelation [YK97]. Their approach was extended to rotational motion by Pao and Kuo [PK03]. Similarly, Rekleitis obtains locally constant motion by considering the Fourier spectrum of a motion-blurred image [Rek96].

The recent user-assisted approach of Jia [Jia07] and the fully automatic approach of Dai and Wu [DW08] are both able to estimate constant velocity motion by formulating a constraint on the alpha channel of the blurred image, shifting the problem from motion estimation to the ill-posed problem of alpha-matte extraction [WC07].

Motion estimation from a single motion-blurred image is also part of blind image deconvolution approaches [KH96]. As blind deconvolution determines the blur-causing motion and the original image from a single, possibly noisy measurement, the problem is highly underdetermined. To simplify computations, motion is often assumed to be spatially invariant [KH96, FSH+06] or at least locally invariant [Lev07, BJNP06]. This is reasonable if, for instance, a translating camera acquires an image of a static scene with negligible depth variation.

Raskar et al. show that for constant linear motion the blind deblurring problem is not only underdetermined but that, even given user-defined motion, image reconstruction is not unique due to the complete loss of high frequencies [RAT06]. They propose a modified shutter system to record also high frequency content of a scene in one multi-exposure image. Other deconvolution approaches use additional images to gain information about the underlying motion as well as on the frequencies suppressed by the blur: Tico and Vehvilainen use pairs of blurred

and noisy images to determine a spatially invariant blur kernel after image registration [TV06]. Yuan et al. [YSQS07] and Lim and Silverstein [LS08b] assume small offsets between the blurred and the noisy image and include them into the spatially invariant blur kernel estimation. Additionally, they use the noisy image to reduce ringing artifacts during deconvolution. The hybrid camera of Ben-Esra and Nayar acquires a long-exposed image of the scene, while a detector with lower spatial and higher temporal resolution acquires a sequence of short-exposed images [BEN04]. From this sequence they detect camera motion by calculating optical flow between successive images with the iterative algorithm of Lucas and Kanade [LK81]. From the camera motion, a global blur kernel is reconstructed which is used to deblur the image. A recent extension of the hybrid camera permits the kernel to be a local mixture of predefined basis kernels, which can be handled by modern deblurring methods [TDBL08].

The deconvolution approaches of Rav-Acha and Peleg use two motion-blurred images with spatially invariant linear motion-blurs in different directions to obtain improved deconvolution results [RAP00, RAP05]. However, for a dynamic scene and a static camera, different motion-blur directions are hard to obtain. The motion-from-smear approach of Chen et al. [CNM96, CNM95] as well as the approaches of Favaro and Soatto [FS04] and Agrawal et al. [AXR09] therefore employ images with different degrees of motion-blur, i.e., different exposure times, making different simplifying assumptions about the motion. These assumptions range from constant motion [CNM96] over object-wise constant motion [CNM95, FS04] to motion computable from neighboring frames with the same exposure time [AXR09]. Pixelwise varying motion and occlusion are not considered.

By using motion-blurred images from a long-exposure video with constant exposure time, Tull and Kataloggos [TK96] estimate inter-frame and intra-frame motion in a unified approach. Bar et al. [BBRS07] determine constant motion and a segmentation into moving foreground and static background. Both methods are limited by the loss of high frequency content of moving objects.

In all aforementioned approaches, directly or indirectly, the prime goal of the motion estimation is the reconstruction of the original, sharp image. Being just an intermediate step, motion is approximated with simple models and if the quality of the estimated motion is evaluated, it is validated via deconvolution. In our approach, instead, we are interested in recovering high-quality, dense motion fields that may vary from pixel to pixel and that are accurate enough to be used for a broad range of applications. In addition, we are interested in adequate motion estimates also for occluded points and a well-founded estimate of occlusion timings. The advantages of occlusion handling and occlusion timings in image interpolation are demonstrated by Mahajan et al. [MHM+09] using a path-based approach. However, in this approach paths are calculated between two short-exposure images based on a discrete optimization framework, yielding only full pixel accuracy.

Before we look more closely at motion estimation from alternate exposure imaging, we analyze the sources of temporal aliasing in optical flow estimation in the next section. More precisely, we show that blurred object boundaries in long-exposure images and temporal aliasing in short-exposure image optical flow are problems inherent to the recording modalities of the images.

2.3 Signal Theoretic Background

Acquiring videos of moving scenes represents a sampling in the spatial as well as in the temporal domain. From sampling theory it is well known that regularly sampling a signal at a frequency below its Nyquist limit leads to aliasing artifacts [Mar91]. In the case of temporal aliasing, this effect is also known as the *wagon-wheel-effect*. In this section, the occurrence of temporal aliasing in two-image optical flow and the filtering properties of long exposure times are analyzed. Based on this analysis the short-comings of using only short-exposure or only long-exposure images for motion estimation become obvious.

For the analysis it is sufficient to consider the easiest case of a spatially band-limited image $f_0(x, y)$ that moves with uniform constant velocity $\mathbf{w} = \binom{u}{v}$. Then for every instant $t \in \mathbb{R}$ the image f on the sensor plane can be expressed as

$$f(x, y, t) = f_0(x - ut, y - vt) \ . \tag{2.1}$$

Aliasing effects become apparent in the Fourier domain. The 2D Fourier transform of $f_0(x, y)$ is denoted as $F_0(\xi, \eta)$. Under the assumption that F_0 is band-limited, a pointwise spatial sampling distance can be chosen so that the spatial replica are well separated. The 3D Fourier transform $F(\xi, \eta, \tau)$ of the moving image is related to the 2D Fourier transform $F_0(\xi, \eta)$ of the static image via

$$\begin{aligned} F(\xi, \eta, \tau) &= \iiint f(x, y, t) e^{-2\pi i(\xi x + \eta y + \tau t)} \, dx \, dy \, dt \\ &= \iiint f_0(\tilde{x}, \tilde{y}) e^{-2\pi i(\xi \tilde{x} + \eta \tilde{y})} \, e^{-2\pi i(\tau + \xi u + \eta v)t} \, d\tilde{x} \, d\tilde{y} \, dt \\ &= F_0(\xi, \eta) \, \delta(\tau + \xi u + \eta v) \end{aligned} \tag{2.2}$$

where δ is the Dirac delta. While the support of F_0 is in the $(\tau = 0)$-plane, the support of the space-time Fourier transform of the uniformly moving image is located on the hyperplane $\binom{u}{v}^{\perp}$, i.e., the support is normal to the motion direction. Additionally, the support is stretched according to motion magnitude, Fig. 2.2.

Pointwise temporal sampling leads to replica of the original transform on the τ axis in a distance inverse to the sampling distance. Motion that is larger than the inverse of twice the spatial band-limit per frame, i.e., motion larger than one pixel per frame, shears the support of the 3D Fourier transform up to the point where the projections of the supports of the replica to the ($\xi = 0$, $\eta = 0$)-subspace, i.e., the τ axis overlap, Fig. 2.3.

Considering a fixed point (x_0, y_0), the temporal intensity function for this point, $\phi(t) = f(x_0, y_0, t)$, has the temporal Fourier transform

$$\begin{aligned}
\Phi(\tau) &= \int e^{-2\pi i\tau t} f_0(x_0 - tu, y_0 - tv)\ dt \qquad (2.3)\\
&= \iint \delta(y - (y_0 - tv))\ e^{-2\pi i\tau t}\ f_0(x_0 - tu, y)\ dy\ dt\\
&= \iiint e^{-2\pi i(\eta(y-y_0+tv)+\tau t)}\ f_0(x_0 - tu, y)\ dy\ dt\ d\eta.
\end{aligned}$$

Substituting $x = x_0 - tu \Leftrightarrow t = \dfrac{x_0 - x}{u}$ for $u \neq 0$ gives

$$\begin{aligned}
\Phi(\tau) &= \frac{-1}{|u|} \iiint e^{-2\pi i(\eta(y+\frac{x_0-x}{u}v-y_0)+\tau\frac{x_0-x}{u})}\ f_0(x, y)\ dy\ dx\ d\eta \qquad (2.4)\\
&= \frac{-1}{|u|} \iiiint e^{-2\pi i(\eta y+\xi x)}\ e^{2\pi i(\eta y_0+\xi x_0)}\ \delta(\xi + \frac{\tau + \eta v}{u})\ f_0(x, y)\ dy\ dx\ d\eta\ d\xi\\
&= -\iint e^{2\pi i(\eta y_0+\xi x_0)}\ \delta(\tau + \xi u + \eta v)\ F_0(\xi, \eta)\ d\xi\ d\eta\\
&= -\iint e^{2\pi i(\eta y_0+\xi x_0)}\ F(\xi, \eta, \tau)\ d\xi\ d\eta.
\end{aligned}$$

In other words, $\Phi(\tau)$ is a weighted projection of $F(\xi, \eta, \tau)$ to the ($\xi = 0$, $\eta = 0$)-subspace.

The Fourier transform of the temporal derivative of the intensity function, $\frac{d}{dt}\phi(t)$ can be obtained from $\Phi(\tau)$ by multiplication with $2\pi i\tau$. When the motion is large, the projections to the ($\xi = 0$, $\eta = 0$)-subspace overlap and therefore introduce aliasing into the temporal Fourier transform as well as into the transform of the temporal derivative. Yet, temporal derivatives are used in all optical flow algorithms based on the color constancy assumption, Eq. (1.3).

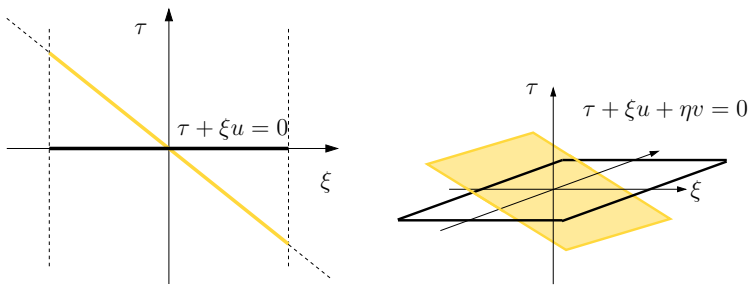


Figure 2.2: (a) Static 1D or (b) 2D signals have space-time Fourier transforms with support on the ($\tau = 0$) hyperplane (black). The Fourier transform of a uniformly moving signal has a sheared support that lies on the motion-dependent hyperplane $\tau + \xi u = 0$ or $\tau + \xi u + \eta v = 0$ respectively (yellow).

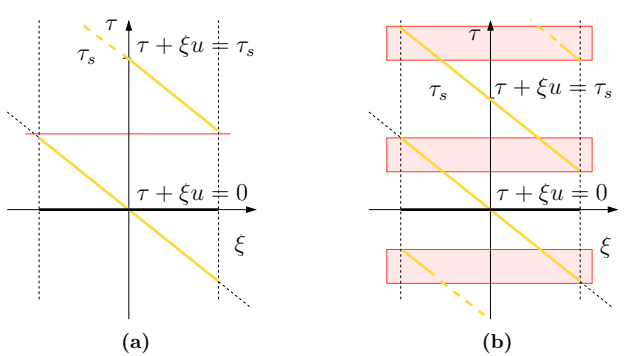
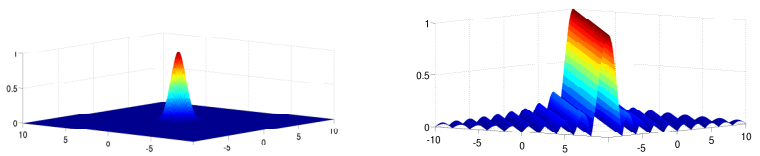


Figure 2.3: Regular temporal sampling of a moving 1D signal with sampling frequency τ_s leads to replica in the space-time Fourier domain. (a) If motion is smaller than one pixel per frame, the projections of the replica to the τ axis don't overlap. (b) For larger motion the projections of the replica on the τ axis do overlap (boxes) causing aliasing artifacts if used without adequate filtering.



(a) **(b)**

Figure 2.4: (a) The isotropic spatial filter used in multi-scale optical flow algorithms attenuates high frequencies in all directions indiscriminately. (b) Long-exposure imaging corresponds to filtering the image with an anisotropic filter that correctly attenuates high frequencies only in the direction where temporal aliasing can occur due to the underlying motion. Frequencies perpendicular to this direction are left unchanged.

To reduce the overlap of the replica, several approaches are employed. The straight-forward approach is to increase the temporal sampling rate. High-speed cameras provide high temporal sampling rates, and the optical flow can be determined between consecutive images [LAG05]. We compare the results of this approach to motion fields estimated with alternate exposure images in Sect. 2.7.

If no high-speed video equipment is available, other methods have to be applied to avoid aliasing. In multi-scale optical flow approaches, the entire image is low-pass filtered isotropically in both spatial directions, Fig. 2.4(a), to indiscriminately remove high spatial frequencies [Chr00]. While this approach does remove the frequencies that cause aliasing, it also destroys high frequency information that is not affected by aliasing at all and can be used to limit the solution space of the motion estimation problem.

A pre-filtering restricted to the direction in which aliasing actually occurs can be obtained by using longer exposure times. Approximating the shutter function with the boxcar-function

$$h(x, y, t) = \begin{cases} 1 & \text{if } |t| \leq \frac{T}{2} \\ 0 & \text{else} \end{cases} \tag{2.5}$$

where T is the total duration of the exposure, a long-exposure image

$$g(x, y, t) = (f * h)(x, y, t) \tag{2.6}$$

has the 3D Fourier transform

$$\begin{aligned} G(\xi, \eta, \tau) &= \iiint \int_{-\frac{T}{2}}^{\frac{T}{2}} f(x, y, t+s)\, ds\, e^{-2\pi i(\xi x + \eta y + \tau t)} dx\, dy\, dt \\ &= F_0(\xi, \eta)\, T sinc(\pi T(\xi u + \eta v))\, \delta(\tau + \xi u + \eta v)\ . \end{aligned} \tag{2.7}$$

In consequence, the high frequencies of the original image are low-pass filtered in the direction of the motion, Fig. 2.4(b), and left unchanged in the direction perpendicular to it.

Although attenuating the high frequencies in motion direction is ideally suited for the consideration of temporal derivatives, the loss of these frequencies is a drawback for users interested in the video's content rather than motion estimation. Due to the zeros of the *sinc* function, some frequencies are irreversibly destroyed [RAT06, AXR09] while other frequencies are strongly damped and so their reconstruction is heavily susceptible to noise [KH96].

Considering only motion-blurred images for traditional two-image motion estimation, the occlusion problem is further emphasized. In short-exposure images only the occluded pixels have no correspondence in the subsequent image while for occluding pixel a correspondence exists. For long-exposure images, foreground and background pixels mix at occlusion boundaries and so all pixels concerned with occlusion in either of the images have no correspondence.

In our approach we use short- and long-exposure images that are recorded alternatingly. While the short-exposure images provide the information about high spatial frequencies, the long-exposure images provide the information about the motion, i.e., a measurement how motion combines the pixels of the short-exposure images. Temporal derivatives are no longer required and high frequencies perpendicular to the motion can assist in motion detection. In the next section, a suitable image formation is derived, which also models occlusion.

2.4 Image Formation Model

In order to exploit the information provided by the additional long-exposure image, we need an image formation model that relates the acquired images via a dense 2D motion field. As input, we assume two images $I_1, I_2 : \Omega \to \mathbb{R}$ with short exposure times which are taken before and after the exposure time of a third, long-exposure input image $I_B : \Omega \to \mathbb{R}$. We look for an image formation model that describes a motion-blurred image $B : \Omega \to \mathbb{R}$ in terms of I_1 and I_2 and the unknown motion; a model that can be compared to the actual measurement I_B.

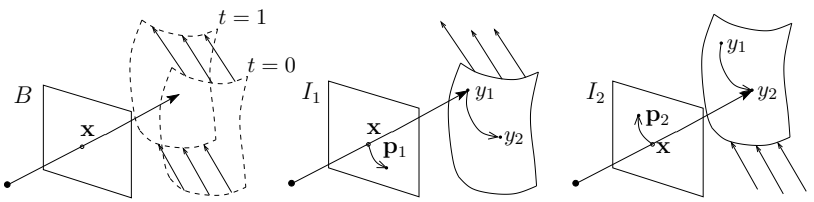
Some additional assumptions are made. We assume that the short-exposure images are free of motion-blur and that short-exposure and long-exposure images are brightness-adjusted such that in case of no motion, all images are identical. In practice, we adjust the gain factor of the camera according to the exposure duration. Finally, we assume that scene surface appearance does not change considerably between the exposure time of all three images.

2.4.1 Without Occlusion

Our goal is to derive a suitable model for the formation of the motion-blurred image B, which is both computationally manageable as well as sufficiently accurate to describe real-world data. For the simplest case, let us consider a moving scene without any occluded or disoccluded scene points, which implies that all scene points contributing to the motion-blurred image B are visible in I_1 as well as I_2. Parameterizing by time $t \in [0, 1]$ we obtain

$$B(\mathbf{x}) = \int_0^1 I_1(\mathbf{p}_1(\mathbf{x}, t))\ dt = \int_0^1 I_2(\mathbf{p}_2(\mathbf{x}, t))\ dt. \tag{2.8}$$

where $\mathbf{p}_1(\mathbf{x}, \diamond) : [0, 1] \to \Omega$ and $\mathbf{p}_2(\mathbf{x}, \diamond) : [0, 1] \to \Omega$ are spatially varying, planar curves on the image plane with $\mathbf{p}_1(\mathbf{x}, 0) = \mathbf{x}$ and $\mathbf{p}_2(\mathbf{x}, 0) = \mathbf{x}$, Fig. 2.5. For each input image, the curves describe the points on the image plane which pass through $\mathbf{x}$ during the exposure duration. While $\mathbf{p}_1$ orders the points forward in



(a) $\forall\, t \in [0,1]$ **(b)** $t = 0$ **(c)** $t = 1$

Figure 2.5: (a) During the exposure time of the long-exposure image B multiple scene points contribute to a pixel $\mathbf{x}$ on the image plane. (b) Without occlusion, these scene points form a contiguous path with endpoints y_1 and y_2 on the 3D surface. The projection of the 3D path onto the image plane for the preceding short-exposure image I_1 is a planar curve $\mathbf{p}_1$. (c) Projecting the 3D path to the image plane for the succeeding short-exposure image I_2 yields a planar curve $\mathbf{p}_2$.

time, $\mathbf{p}_2$ orders them backward in time. In the case without occlusion, the entire curves are visible in both images, so that the values of both integrals are equal.

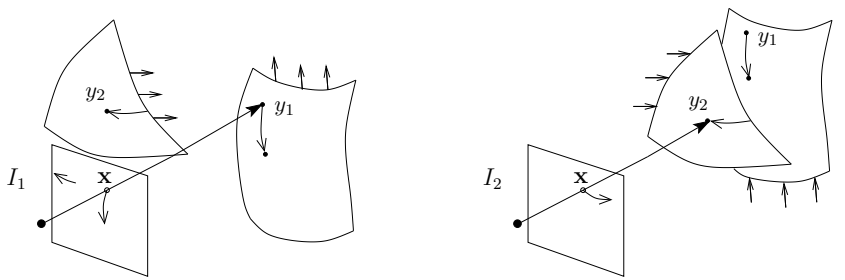
2.4.2 With Occlusion

The long-exposure image enables incorporating occlusion effects into the image formation model. We assume that a point changes its visibility at most once during the exposure. If a scene surface becomes occluded, some parts of the motion paths are visible in only one of the two short-exposure images, Fig. 2.6. We partition the integral so that part of the intensity $B(\mathbf{x})$ observed in $\mathbf{x}$ is due to intensities along curve $\mathbf{p}_1$, while the remaining part is due to intensities along $\mathbf{p}_2$,

$$B(\mathbf{x}) = \int_0^{s(\mathbf{x})} I_1(\mathbf{p}_1(\mathbf{x},t))\; dt + \int_0^{1-s(\mathbf{x})} I_2(\mathbf{p}_2(\mathbf{x},t))\; dt. \tag{2.9}$$

Here, $s(\mathbf{x}) \in [0,1]$ denotes the moment during exposure where an object previously visible at position $\mathbf{x}$ in I_1 becomes occluded by an object visible at $\mathbf{x}$ in I_2, or vice versa.

Note that in the case of no occlusion, any choice of s yields the same intensity $B(\mathbf{x})$. The occlusion timings are only well defined in areas where occlusion



(a) $t = 0$ **(b)** $t = 1$

Figure 2.6: With occlusion, the path of scene surface points contributing to pixel $\mathbf{x}$ in the long-exposure image is split into two parts. The first part is on the occluded, the second part is on the occluding surface. (a) The path has a non-contiguous projection to the image plane of the preceding image I_1 and (b) is only partly visible in the succeeding image I_2.

actually takes place. At all other points any value $s \in [0, 1]$ is equally valid. If we consider a fixed, non-occluded image point $\mathbf{x}$ and differentiate (2.9) with respect to s, we obtain the brightness constancy assumption of traditional optical flow computation expressed by the blurred image's motion curves

$$0 = I_1(\mathbf{p}_1(\mathbf{x}, s)) - I_2(\mathbf{p}_2(\mathbf{x}, 1 - s)) \; . \tag{2.10}$$

Thus our image formation model can be considered as a generalization of the brightness constancy assumption, Eq. (1.1). In contrast to the brightness constancy assumption our image formation model incorporates the additional information provided by the motion-blurred image and explicitly takes occlusion into account.

The image formation model can be easily extended to allow for more than one visibility change, given that all passing pixels are visible either in I_1 or I_2. However, for the sake of stability of computations and since multiple occlusions arise only rarely in practical situations with reasonable frame rates, we do not further discuss this extension.

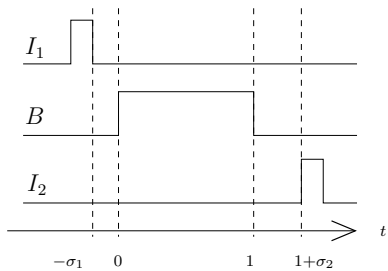


Figure 2.7: We include a temporal offset to account for gaps that may occur due to recording hardware constraints. Scaling the exposure time of B to unit length, the gap between I_1 and B has duration σ_1, and the gap between I_2 and B has duration σ_2.

2.4.3 With Temporal Offset

We want to allow for exposure gaps between the images I_1 and B as well as between B and I_2, Fig. 2.7. Gaps between exposures occur, e.g., due to camera hardware constraints. Scene motion, of course, continues during such exposure gaps. To account for gaps, we include a temporal offset in Eq. (2.9) by changing the integration limits corresponding to the relative lengths of the gaps:

$$B(\mathbf{x}) = \int_{\sigma_1}^{\sigma_1+s(\mathbf{x})} I_1(\mathbf{p}_1(\mathbf{x},t))\ dt\ + \int_{\sigma_2}^{\sigma_2+1-s(\mathbf{x})} I_2(\mathbf{p}_2(\mathbf{x},t))\ dt\ . \tag{2.11}$$

σ_1 is the quotient of the length of the exposure gap between I_1 and B and the exposure duration of B. Correspondingly, σ_2 is the quotient of the length of the exposure gap between B and I_2 and the exposure duration of B. As before $s(x) \in [0,1]$ is the moment of occlusion.

2.4.4 Frame Interpolation

The motion curves $\mathbf{p}_1$ and $\mathbf{p}_2$ describe which points from I_1 and I_2 pass at an image point $\mathbf{x} \in \Omega$ during the exposure time and sum up to the motion-blurred image. We can evaluate the motion paths for a fixed $t \in [0,1]$ and obtain an

intermediate frame I_t. Setting

$$I_t(\mathbf{x}) = \begin{cases} I_1(\mathbf{p}_1(\mathbf{x}, t)) & \text{if } t \leq s(\mathbf{x}) \\ I_2(\mathbf{p}_2(\mathbf{x}, 1-t)) & \text{if } t > s(\mathbf{x}). \end{cases} \tag{2.12}$$

even occluded and disoccluded points are interpolated correctly without the need for explicit occlusion detection.

2.4.5 From Motion Curves to Displacement Vectors

Motion curves $\mathbf{p}_1$ and $\mathbf{p}_2$ describe the motion in the coordinate frame of the motion-blurred image. Since for many applications a forward or backward motion field is needed, we warp the motion curves according to the estimated motion and occlusion parameters to obtain a displacement field for I_1 and I_2, respectively. For every pixel in the motion-blurred image we follow the estimated motion curve to the latest admissible pixel until its occlusion time. For a pixel in the short-exposure image, we than average between all assigned motion curves or interpolate bilinearly if necessary.

In the image formation model described so far, we used general motion curves. To simplify computations and obtain a parameterization with the minimum number of unknowns, we adopt here a linear motion model so that

$$\mathbf{p}_1(\mathbf{x}, t) = \mathbf{x} - t\,\mathbf{w}_1(\mathbf{x}) \;\text{ and }\; \mathbf{p}_2(\mathbf{x}, t) = \mathbf{x} + t\,\mathbf{w}_2(\mathbf{x}), \tag{2.13}$$

where $\mathbf{w}_j : \Omega \to \mathbb{R}^2$, $\mathbf{w}_j(\mathbf{x}) = \binom{w_{j,1}(\mathbf{x})}{w_{j,2}(\mathbf{x})}$ for $j \in \{1, 2\}$. Assuming constant velocity during the whole exposure interval is more restrictive than the local linearity assumption in warping based optical flow computation but it turns out to be a suitable approximation also for more general types of motion, Sect. 2.7. If desired, however, it is straight-forward to extend the algorithm to more complex pointwise motion models.

2.5 Least Squares Approach

The image formation model for a motion-blurred image B considered in the previous section yields a pointwise error measure for estimates of the motion paths. Given two short-exposure images I_1, I_2 and a long-exposure image I_B, i.e., the actual measurement, we can compare the blurred image I_B to the result B predicted by the model (2.11):

$$e(\mathbf{x}, \mathbf{w}_1, \mathbf{w}_2, s) = B(\mathbf{x}, \mathbf{w}_1, \mathbf{w}_2, s) - I_B(\mathbf{x}). \tag{2.14}$$

In this error measure there are 5 unknowns for every pixel $\mathbf{x}$ in the image domain, i.e., horizontal and vertical component for the forward as well as the backward motion path and the occlusion time. The minimization of e with respect to these variables can have several equally valid solutions, e.g., by letting $s = 0$ for an unoccluded point the backward motion path $\mathbf{w}_2$ can be chosen arbitrarily. Thus the problem is ill-posed in the sense of Hadamard [Had02]. In the next section we give a first approach to make the problem computationally manageable by introducing additional assumptions. A different set of assumptions, which allows for a global energy formulation, is introduced and discussed in Sect. 2.6.

2.5.1 Additional Assumptions

In order to reduce the number of unknowns in the energy formulation, we first consider a point that is neither occluded nor disoccluded during the exposure interval. That is, the long-exposure image acquires its gray value for that point from only one object. It is reasonable to assume that motion within one object changes only slightly, so that we can approximate the forward and backward paths to be equal $\mathbf{w}_1 \approx \mathbf{w} \approx \mathbf{w}_2$. This approximation holds exactly for a rigid translation parallel to the image plane. To test the robustness of the algorithm towards this assumption, we evaluate the algorithm also for scenes where it is violated, Sect. 2.7.

For a non-occluded point all occlusion times s are equally valid so we can additionally evaluate the integral for a fixed sequence $0 \leq s_1 < \ldots < s_N \leq 1$. Fixing the occlusion times not only renders the estimation of s superfluous, but also provides us with N equations, each contributing to find the correct motion path, Sect. 2.5.2.

The above two steps reduce the number of unknowns per pixel to the two components of the motion path vector $\mathbf{w}$ and increase the number of non-linear equations to N.

If a point is occluded, forward and backward motion differ. Thus optimization under the assumption $\mathbf{w}_1 \approx \mathbf{w} \approx \mathbf{w}_2$ is expected to lead to a comparably high residual. Only for points with high residual, we assume different forward and backward motion paths. To enable computation of the occlusion time - a crucial variable for occluded points - the assumption of locally constant motion paths is made, so that the motion information can be inferred from neighboring non-occluded pixels.

Applying the above assumptions, we now consider the resulting optimization problem and its solution more specifically. An overview of the resulting algorithm is shown in Fig. 2.8.

2.5.2 Pointwise Optimization Problem

With the assumption $\mathbf{w}_1 \approx \mathbf{w} \approx \mathbf{w}_2$ introduced in the previous section, for the fixed sequence $0 \leq s_1 < \ldots < s_N \leq 1$ and for each $i \in \{1, \ldots, N\}$ we consider

$$F_i(\mathbf{x}, \mathbf{w}) = \left(I_B(\mathbf{x}) - \int_{\sigma_1}^{\sigma_1 + s_i} I_1(\mathbf{x} - t\mathbf{w})\, dt - \int_{\sigma_2}^{\sigma_2 + 1 - s_i} I_2(\mathbf{x} + t\mathbf{w})\, dt \right)^2 , \quad (2.15)$$

i.e., for a fixed point $\mathbf{x} \in \Omega$, F_i describes the deviation of the measured motion-blurred image from the model value for a given motion path $\mathbf{w} \in \mathbb{R}^2$ using the differentiable squared distance. If all the assumptions hold exactly, $F_i = 0$ for the true motion path and for all $i \in \{1, \ldots, N\}$. Note that the split in the integral at s_i permits to restrict the solution space. Consider for example an image with

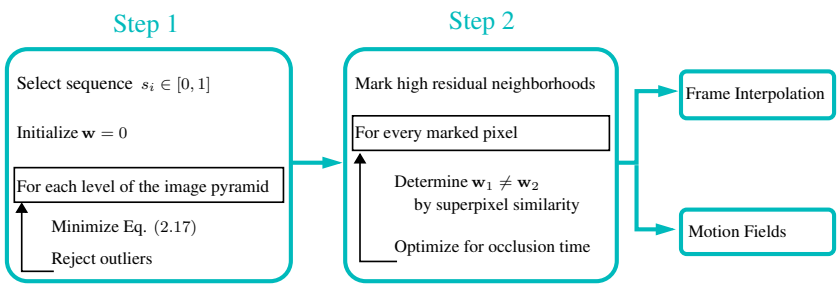


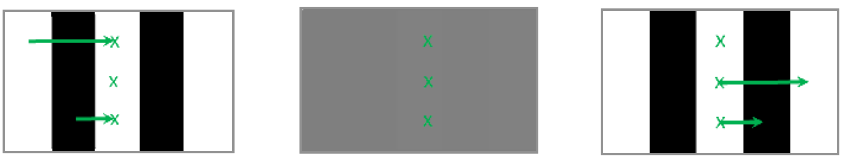
Figure 2.8: The workflow of the least squares approach assumes forward and backward motion paths to be symmetric in the first step. Only in the second step the possibility of occlusion is considered for points with a high residual. With the motion paths and occlusion timings, images can be interpolated directly, or traditional motion vector fields for each pixel in the short-exposure images can be determined.

black and white stripes of width d that moves exactly twice the distance d in horizontal direction, Fig. 2.9. Letting only $s = 0$ or $s = 1$ the motion could be $2d$, Fig. 2.9(a), as well as d, Fig. 2.9(b). Requiring $F_i = 0$ also for intermediate values of s, e.g. $s_i = \frac{1}{2}$, excludes the latter solution, Fig. 2.9(b) last row. Similar examples hold for the determination of the path's angle. At the limit $N \to \infty$ only the true motion path can satisfy all $F_i = 0$ simultaneously for an arbitrary image. Interestingly, given only the images I_1 and I_2 in Fig. 2.9, a two-image optical flow algorithm is not able to detect any motion at all while zero motion is not a valid solution if the motion blurred image is taken into consideration.

As increasing the number N of samples for s also increases the amount of computation, we keep N small, e.g. $N = 5$, and additionally include the differentiated version (2.10) for $s = 0.5$ as

$$F_{N+1}(\mathbf{x}, \mathbf{w}) = \left(I_1(\mathbf{x} - (\frac{1}{2} + \sigma_1)\mathbf{w}) - I_2(\mathbf{x} + (\frac{1}{2} + \sigma_2)\mathbf{w}) \right)^2 , \tag{2.16}$$

with $F_{N+1} = 0$ for the true motion path.



I_1 I_B I_2

(a) For the correct motion path, all occlusion timings s provide the correct value for the motion-blurred image I_B.

I_1 I_B I_2

(b) For a wrong motion path, some switching instants s might provide the correct values for the motion-blurred image I_B (two top rows), but not all do (last row).

Figure 2.9: For some occlusion timings s, several different motion vectors might solve the image formation model. Applying a number of different values for s in Eq. (2.15) resolves the ambiguity.

We now try to find a $\mathbf{w} \in \mathbb{R}^2$ that minimizes the pointwise energy

$$E_{LS}(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N+1} F_i(\mathbf{x}, \mathbf{w}) \ . \tag{2.17}$$

Dennis and Schnabel [DS83] describe several numerical methods to solve this non-linear least squares problem. We use a model-trust region implementation of the well-known Levenberg-Marquardt algorithm because of its robustness and reasonable speed. The path integral over the images is calculated using linear interpolation for the image functions I_1 and I_2. The derivatives of the function $F = (F_1, \ldots, F_{N+1})^T$ are determined numerically. In order to attenuate the impact of local noise, we iterate the optimization and smooth intermediate results by replacing motion paths differing more than 0.25 pixels from the motion paths of the majority of its 8 neighbors by the average motion path of the majority.

2.5.3 Multi-Scale Approach

The image formation model of Sect. 2.4 gives rise to a non-linear optimization problem for every pixel. Unfortunately, the best studied algorithms for solving these problems provide only local optima. But we are interested in the global optimum. Therefore, we apply a multi-scale approach. Multi-scale approaches are very common in optical flow computation [BBPW04, ZPB07, XCS+06], and, as illustrated in Sect. 2.3, in the two-image approach necessary to avoid temporal aliasing. In our case, the multi-scale approach is only required for the non-linear optimization and not from the signal theoretic point of view.

For spatially varying motion paths $\mathbf{w}(\mathbf{x})$, motion-blurred image formation and low-pass filtering are not commutative. Yet, assuming $\mathbf{w}(\mathbf{x})$ to be constant on the support of the filter we can easily show the commutativity. For conciseness, we demonstrate the manipulations for one image and zero temporal offset only, but the result transfers to all expressions for B in Sect. 2.4.
Let $G : \mathbb{R}^2 \to \mathbb{R}$ be a linear low-pass filter, then

$$\begin{aligned} & \int_0^1 (G * I_1)(\mathbf{x} - t\mathbf{w})\ dt \qquad (2.18) \\ = & \int_0^1 \int_{\mathbb{R}^2} G(\mathbf{z}) I_1(\mathbf{x} - t\mathbf{w} - \mathbf{z})\ d\mathbf{z}\ dt \\ = & \int_0^1 \int_{\mathbb{R}^2} G(\mathbf{x} - \mathbf{y}) I_1(\mathbf{y} - t\mathbf{w})\ d\mathbf{y}\ dt \\ = & \int_{\mathbb{R}^2} G(\mathbf{x} - \mathbf{y}) \left(\int_0^1 I_1(\mathbf{y} - t\mathbf{w})\ dt \right)\ d\mathbf{y} \\ = & (G * B)(\mathbf{x}) \end{aligned}$$

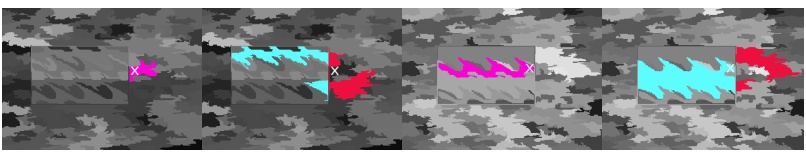
In multi-scale approaches, the algorithm starts at a coarse level and successively refines the solution on finer resolutions. Inaccuracies which are introduced by the assumption of local constancy can be corrected at later stages. Apart from leading the optimization to a global minimum, the multi-scale approach is also able to infer coarse scale motion estimates into large uniform areas where the motion of the brightness pattern is not apparent.

2.5.4 Occlusion

In the occluded regions of the blurred image, components of different motion paths mix and the simplifying assumption of equal forward and backward motion paths $\mathbf{w}_1 \approx \mathbf{w} \approx \mathbf{w}_2$ no longer holds. In occluded regions we expect the pointwise energy E_{LS} in Eq. (2.17) to remain high after optimization. We therefore mark a pixel and its immediate eight neighbors as possibly occluded, if E_{LS} exceeds a threshold T_E. The threshold can be chosen conservatively as points that are erroneously marked occluded will still have their appropriate motion paths assigned. Only their initial estimate is disregarded in the estimation of the foreground and background motion paths. Instead of setting the threshold T_E absolutely and thus also in dependency of N, we choose a percentage of occluded points, e.g. 10%, and set T_E to the corresponding quantile $T_E = Q_{.90}$ of all optimization residuals in the image.

For an occluded/disoccluded pixel, there are two motion paths and the occlusion time necessary to describe the gray value in the blurred image. We circumvent the problem of estimating five variables with one equation for these points by extrapolating the motion paths in the occluded regions from neighboring non-occluded regions. Given estimates for the motion paths, we then determine the occlusion time on the basis of these estimates.

Considering a possibly occluded point we build two clusters C_a and C_b from the motion paths of probably unoccluded points in a neighborhood with a radius of $r = 20$ pixels. With the center of these clusters, we obtain two motion paths. We want to determine which of the two motion paths dominates the integration at the beginning of the exposure interval, i.e., which path integrates the pixels around $\mathbf{x}$ in I_1, and which path dominates the integration at the end of the exposure interval, i.e., which path integrates the pixels around $\mathbf{x}$ in I_2. We use superpixel segmentation [FH04] for this purpose, Fig. 2.10. Let S_{ix} be the superpixel of $I_i(\mathbf{x})$, S_{ia} and S_{ib} the collection of superpixels in I_i containing the pixels that contribute to C_a and C_b respectively and $d(\cdot, \cdot)$ the superpixel distance



(a) (b) (c) (d)

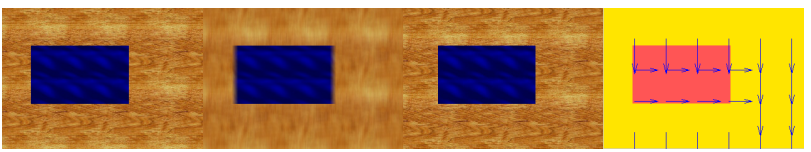
Figure 2.10: We generate a superpixel segmentation of the two short-exposure images, where each superpixel obtains a label represented by a different gray value. For an occluded point x we determine its superpixel (a) in the first short-exposure image and (c) in the second short-exposure image (magenta). We build two clusters from the motion paths of unoccluded points in the neighborhood and compare their superpixels in the short-exposure images, (b) and (d) (blue, red), to the superpixels of x to assign motion paths for the beginning and the end of the exposure interval.

also defined in [FH04]. The superpixel of a disoccluded point and the superpixels containing the foreground motion should belong to the same object in the first short-exposure image and thus the superpixel distance between them is expected be small. The containing superpixels might even be identical. In the second image, the superpixel of the disoccluded point belongs to the background and is therefore expected to be similar or equivalent to the superpixels of the background motion in this image. More generally, if the inequality $d(S_{ix}, S_{ia}) + d(S_{jx}, S_{jb}) < d(S_{ix}, S_{ib}) + d(S_{jx}, S_{ja})$ holds for $i = 1$ and $j = 2$ or for $i = 2$ and $j = 1$ we assign the motion of C_a to $\mathbf{w}_i$ and that of C_b to $\mathbf{w}_j$. Else we deduce that the point is not occluded after all and assign the motion path with the smallest residual in Eq. (2.17).

Given motion paths $\mathbf{w}_1$ and $\mathbf{w}_2$ only the occlusion time s remains to be estimated. We minimize

$$E_s(\mathbf{x}, s) = \left(I_B(\mathbf{x}) - \int_{\sigma_1}^{\sigma_1+s} I_1(\mathbf{x} - t\mathbf{w}_1)\, dt - \int_{\sigma_2}^{\sigma_2+1-s} I_2(\mathbf{x} + t\mathbf{w}_2)\, dt \right)^2 \quad (2.19)$$

by a straightforward line search algorithm as described in [FMM76].



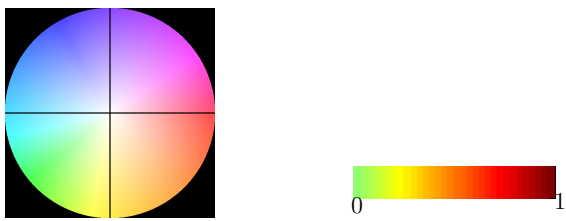
(a) I_1 **(b)** I_B **(c)** I_2 **(d)** $\mathbf{w}_{gt}$

Figure 2.11: (a) - (c) The input images I_1, I_B and I_2 of the synthetic test scene *square* and (d) the ground-truth forward motion field $\mathbf{w}_{gt}$, color-coded with the map in Fig. 2.12(a) and overlaid with sparse and scaled arrows.

2.5.5 Experiments

With the algorithm described above, motion paths and occlusion times can be estimated. In this section we are going to discuss the influence of the parameters of the algorithm, e.g., the choice of s_i and of the occlusion threshold T_E. A comparison to state-of-the-art motion estimation algorithms is postponed to Sect. 2.7 where also the difference to the global minimization algorithm presented in Sect. 2.6 is evaluated. Here the evaluation is performed with a simple synthetic example, where ground-truth motion is known. The basic 320×225 pixel test scene *square* with a foreground that translates 10 pixels horizontally and a background that translates 15 pixels vertically is shown in Fig. 2.11 together with the ground-truth displacement map. In this scene with moving foreground and background, pixels with different, non-zeros velocities mix at the object boundaries. Throughout this work, motion fields are color-coded with the qualitative color map shown in Fig. 2.12(a) and overlaid with a sparse arrow field that is scaled for better visibility. Where determined, occlusion timings are color-coded as shown in Fig. 2.12(b) and else set to dark blue.

In Fig. 2.13(a) the result of the first step under the assumption that all points are unoccluded is shown, assigning arbitrary motion paths where points are occluded contrary to the assumptions of the first step. The logarithm of the remaining optimization residual, Fig.2.13(b), is high for points where this assumption is violated. Thresholding the residual is able to remove some of the outliers due to



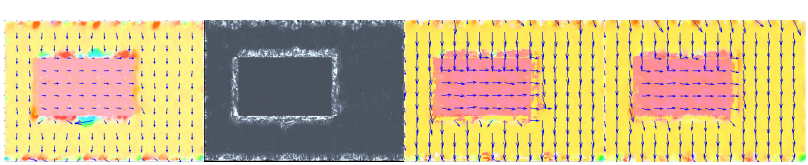
(a) (b)

Figure 2.12: (a) In this work, motion fields are encoded by colors for direction and intensity for magnitude. (b) Where defined, occlusion timings are encoded with a continuous scale between green for $s = 0$ and red for $s = 1$.

occlusion, Fig 2.13(c). The forward displacement field for every pixel in the first short-exposure image is then obtained by warping the motion paths, Fig. 2.13(d).

As different settings for s_i and T_E result in visually very similar motion fields, we evaluate the average angular error (AAE) and the average endpoint error (AEE) [BSL+07] to measure the impact of the parameter.

In the first experiment, we vary the number N of intermediate values for s while keeping all other parameters fixed, i.e., using a 6 level image pyramid, 3



(a) (b) (c) (d)

Figure 2.13: (a) Motion paths are estimated under the assumption of symmetric forward and backward paths, here with the standard parameter $N = 5$. (b) Where the assumption is violated the log of the optimization residual is high (white). (c) After occlusion detection with the parameter value $T_E = Q_{.90}$, forward motion paths for the occluded areas are extrapolated by superpixel comparison. (d) The forward displacement field shown is obtained by warping the motion paths in (c).

N	2	3	4	5	6	7	8	9	10
time [sec]	7529	7612	7621	7797	7846	7912	8065	8139	8180
AAE [°]	7.81	6.78	6.82	6.24	6.68	6.62	6.50	6.49	6.51
AEE [px]	2.28	1.90	1.85	1.73	1.82	1.81	1.77	1.74	1.76

Table 2.1: Increasing the number N of equi-distant intermediate values for the occlusion times s also increases the computation time (3.06 GHz processor, non-optimized, pointwise MATLAB code). Fixing the threshold for occlusion detection $T_E = Q_{.90}$, the smallest average angular error (AAE) and the smallest average endpoint error (AEE) are obtained for $N = 5$.

iterations on each scale and an outlier threshold of 0.25 pixels. To obtain optimal cover for any length of the motion paths, we distribute the s_i equally in the interval $[0, 1]$, i.e., $s_i = \frac{i-1}{N-1}$ for $i \in \{1, \ldots, N\}$.

If the number N of the equidistant intermediate values for s is chosen larger than 2 it has only a small influence on the resulting error, Tab. 2.1. Also, as the optimization implementation works with a minimum of function evaluations, the impact of N on the total computation time is small compared to the total computation time. Apart from determining the number and in our setup the spacing of the s_i the number N also influences the weight of the color constancy assumption in F_{N+1}. The larger N the smaller is the weight of F_{N+1} among all equations. As a trade-off between the equations $F_i, i \in \{1 \ldots N\}$ based on the motion-blurred image and the equation F_{N+1} based on the short-exposure images, $N = 5$ results in the smallest angular error and the smallest endpoint error.

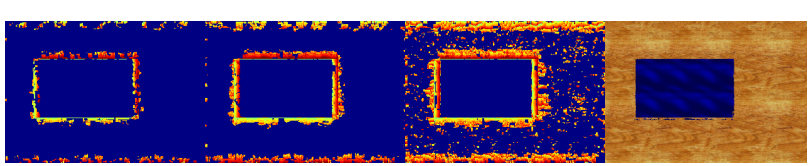
In the next experiment, we consider the threshold T_E. For a fixed number $N = 5$ of intermediate values for s, we change the number of points that are considered as occluded by setting T_E to the corresponding quantile. Considering up to 30% of the pixels as occluded has only a small impact on the AAE and AEE, Tab. 2.2. If more than 30% of the pixels are assigned the occluded label, the AAE and the AEE increases more drastically. Looking at Fig. 2.14, the reason for this becomes obvious: as T_E takes smaller values, more and more points are labeled

T_E	$Q_{.95}$	$Q_{.90}$	$Q_{.85}$	$Q_{.80}$	$Q_{.75}$	$Q_{.70}$	$Q_{.65}$	$Q_{.60}$	$Q_{.55}$	$Q_{.50}$
AAE [°]	6.96	6.24	6.90	6.88	6.84	6.83	7.32	7.40	7.75	7.76
AEE [px]	1.96	1.73	1.82	1.80	1.78	1.78	1.91	1.92	2.02	2.04

Table 2.2: Fixing the number $N = 5$ of intermediate values for s, the smallest average angular error (AAE) and the smallest average endpoint error (AEE) are obtained for $T_E = Q_{.90}$, i.e., when considering 90% of the pixels as non-occluded.

as occluded. But not only occluded points are labeled, also unoccluded points with an inaccurate motion field or noisy pixels in the short-exposure image might be labeled as occluded. If in contrast an occluded pixel was assigned an arbitrary motion path that is not correct but minimizes the pointwise error successfully, this point will not be labeled as occluded. Assigning motion fields to the labeled points, occluded points with small residual will play a more important role if more unoccluded points are labeled as occluded, thus corrupting the quality of the motion estimation.

Failure in occlusion detection does not only decrease the accuracy of the estimated flow field. As can be seen in Fig. 2.14(d), it also decreases the quality of frame interpolation at object boundaries.



(a) **(b)** **(c)** **(d)**

Figure 2.14: Thresholding the optimization residual for occlusion detection considers mainly truly occluded points (a) for $T = Q_{.95}$ and (b) for $T = Q_{.90}$ but does not detect all occluded points. (c) Setting $T = Q_{.75}$ considers also many non-occluded points as occluded but still does not detect all occluded points. (d) Failures in occlusion detection at object boundaries (for $T = Q_{.90}$) are also visible in frame interpolation at $t = 0.25$.

2.5.6 Discussion

The approach presented in this section is able to estimate motion paths, forward/backward motion fields and occlusion timings from a set of three alternate exposure images. It makes some additional assumptions on the motion paths but requires no further regularization such as, e.g., a smoothness constraint. The resulting error functional can be evaluated pointwise. In theory, the presented approach does not require a multi-scale approach to cope with large motion, but due to numerical reasons, a multi-scale approach is used to optimize the pointwise error functional. One example of the results that can be obtained is shown in the previous section; more examples and comparisons to state-of-the-art algorithms are given in Sect. 2.7. Although reasonable results can be obtained with the proposed algorithm, there are some drawbacks of the approach: The assumption of equal forward and backward motion paths is actually only satisfied if an object moves parallel to the image plane. Already for a uniform linear motion at a small angle to the image plane this assumption is no longer satisfied. Additionally, motion paths are estimated pointwise, so for textureless areas motion is only inferred from neighboring pixels via the multi-scale approach.

In the presented approach, occlusion is estimated separately after a thresholding step. Some occluded points exhibit a low optimization residual and are detected only with very low thresholds so that hardly any valid points remain for forward and backward motion path estimation. Furthermore, the super-pixel approach assigns forward/backward motion paths pointwise on a binary decision, the motion path assignment of neighboring pixels is not considered. However, an implicit, regularized treatment of occluded points without any thresholding is desirable that can take neighboring pixels into account. Considering the drawbacks of the additional assumptions of Sect. 2.5.1 we are going to introduce an improved optimization scheme in the next section.

2.6 Total Variation Approach

Considering the difference between a recorded motion-blurred image and the blurred image predicted by the image formation model gives a pointwise error measure for the path vectors and the occlusion time. As the solution to this problem is not unique for all image points, additional assumptions were introduced in the last section. Yet, these assumptions impose new restrictions onto the motion. This section considers different, less restrictive assumptions on the motion paths by considering spatial similarity of path vectors and occlusion times.

2.6.1 Additional Assumptions

Pixels in images are in general not uncorrelated. Often, spatially neighboring pixels belong to the same real-world object and therefore exhibit similar properties such as color, texture or motion. For our underdetermined pointwise error function, Eq. (2.14), resulting from the alternate exposure image formation model, we can therefore look for the solution of the pointwise problem that is most similar to the solution of neighboring pixels. We can achieve this by adding a regularization term to the pointwise error functional, Eq. (2.14). Regularization is a typical way to estimate solutions of under-determined problems [TA77] and often applied in optical flow estimation to overcome the aperture problem [BFB94, BSL+07]. For image points belonging to the same 3D objects, the spatial gradient of the motion field is assumed to be small. From all the solutions satisfying the data-term we want to chose the one with the smallest gradient.

Yet, at object boundaries, motion changes abruptly and the spatial gradient of the motion field is large. A great amount of research was conducted to find regularization terms that smooth out undesired outliers within an object and avoid oversmoothing at motion boundaries at the same time [CS05]. As demonstrated in previous work [ZPB07], using the total variation as a regularizer for flow fields yields promising results. The total variation of a differentiable motion field can

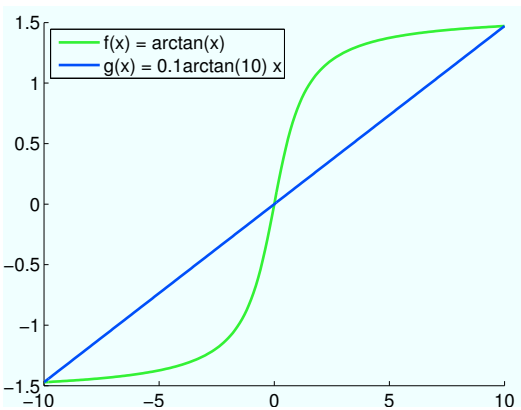


Figure 2.15: While the total variation of the steep function f and the continuously increasing function g are equal, the squared norm of the gradient of f is much higher than that of the gradient of g. Because total variation regularization allows for piecewise constant solutions, we apply it to our motion field estimation, assuming that motion boundaries between different objects are sharp while motion within an object changes slowly, if at all.

be defined as the average norm of the gradient of the motion field

$$TV(\mathbf{w}(\mathbf{x})) = \int_{\Omega} |\nabla \mathbf{w}(\mathbf{x})| \, d\mathbf{x}. \tag{2.20}$$

While the total variation for a steep monotonous function and a smoothly increasing, monotonous function with the same endpoints is the same, the customary squared norm of the gradient punishes large deviations from a constant function much severer than a gradual change, Fig 2.15. Therefore, least squares regularizations of the gradient of the motion field tend to oversmooth motion boundaries, while the total variation regularization of the motion field favors piecewise constant vector fields - which is in accordance with our understanding of only slightly deforming scene objects moving with individual velocities.

In optical flow estimation, the total variation regularization has been further developed by considering anisotropic variants of the total variation, regularizing

the flow only perpendicular to the image gradient and imposing no constraint at all in direction of the gradient [WTP+09]. In short-exposure images, motion boundaries often coincide with high image gradients. Therefore, anisotropic smoothing as a function of the image's gradient can prevent smoothing over object boundaries. In our case of motion path estimation, the underlying image is motion-blurred. In motion-blurred images, high gradients in motion direction are suppressed and not available for anisotropic smoothing. We therefore apply the usual, isotropic total variation here. As the total variation is not differentiable when spatially neighboring motion vectors are very similar, a specialized minimization scheme is applied.

2.6.2 Global Optimization Problem

The central part of the optimization problem is, as before, the pointwise comparison of the recorded motion-blurred image I_B and the result B predicted by the image formation model. We consider the data-term with a robust penalizer $\phi_\epsilon(x) = \sqrt{x^2 + \epsilon}$ where $\epsilon = 10^{-3}$, i.e., we consider

$$G_1(\mathbf{x}, s, \mathbf{w}_1, \mathbf{w}_2) = \phi_\epsilon(B(\mathbf{x}) - I_B(\mathbf{x})). \tag{2.21}$$

Introduced to motion estimation by Black and Anandan [BA96], robust penalizers like ϕ_ϵ are a differentiable version of the absolute value and allow for accurate motion estimation also in the presence of outliers and deviations from the assumptions. Instead of being misled by the strong weight of the quadratic error term of the least squares approach, the absolute value or robust penalizer ϕ_ϵ of the data-term punishes outliers less severely and can thus converge to a good motion estimate even in the presence of deviations from the image formation model [HRRS86].

As in Sect. 2.5.2, we also include the differentiated version and consider it as an additional data-term

$$G_2(\mathbf{x}, s, \mathbf{w}_1, \mathbf{w}_2) = \phi_\epsilon(I_1(\mathbf{x} - (\frac{1}{2} + \sigma_1)\mathbf{w}_1) - I_2(\mathbf{x} + (\frac{1}{2} + \sigma_2)\mathbf{w}_2)) \ . \tag{2.22}$$

Integrating the weighted sum of the pointwise errors over the image domain, we obtain the data-term

$$E_{data}(s, \mathbf{w}_1, \mathbf{w}_2) = \int_\Omega G_1(\mathbf{x}, s, \mathbf{w}_1, \mathbf{w}_2) + \gamma\ G_2(\mathbf{x}, s, \mathbf{w}_1, \mathbf{w}_2)\ d\mathbf{x} \tag{2.23}$$

with $\gamma \geq 0$. Regularizing both path vectors as well as the occlusion time with their total variation results in the final energy functional

$$E_{TV}(s, \mathbf{w}_1, \mathbf{w}_2) = \int_\Omega G_1 + \gamma\ G_2 + \alpha \sum_{i=1}^{2} (|\nabla w_{1,i}| + |\nabla w_{2,i}|) + \beta |\nabla s|\ d\mathbf{x} \tag{2.24}$$

where $\alpha,\ \beta > 0$ are two free parameters of the approach. This energy functional interconnects the pointwise error measure given by G_1 and G_2 via the regularization terms so that now a global minimization is performed. The absolute value considered in the total variation is not differentiable and we therefore adopt a minimization scheme that is presented in the next section.

2.6.3 TV-L^1 Minimization

Our minimization scheme is based on the primal-dual algorithm used for TV-L^1 optical flow [ZPB07], whose variants currently rank in the top of the Middlebury benchmark [BSL+07]. We briefly review the method here and show how we adopt the framework to minimize our more complex energy functional in the next section. We also use the same framework in the multi-view consistent optical flow in section 3.3.

For the very general case of minimizing a total variation energy of the form

$$E(\mathbf{u}) = \lambda \int_\Omega \psi(\rho(\mathbf{u}))\ d\mathbf{x} + \int_\Omega \sum_{i=1}^{k} |\nabla u_i|\ d\mathbf{x} \tag{2.25}$$

for a k-dimensional function $\mathbf{u} = (u_1, \ldots, u_k)^\top$ on Ω with a pointwise error term ρ, an auxiliary vector field $\mathbf{v}$ is introduced and the approximation

$$E_\theta(\mathbf{u}, \mathbf{v}) = \int_\Omega \lambda\ \psi(\rho(\mathbf{v})) + \frac{1}{\theta}|\mathbf{u} - \mathbf{v}|^2 + \sum_{i=1}^{k} |\nabla u_i|\ d\mathbf{x} \tag{2.26}$$

is considered instead. If θ is small, $\mathbf{v}$ will be close to $\mathbf{u}$ near the minimum, and thus E will be close to E_θ. The key result of Ref. [ZPB07] is that Eq. (2.26) can be minimized very efficiently using an alternating scheme that iterates between solving a global minimization problem for each u_i, keeping $\mathbf{v}$ fixed

$$\min_{u_i} \int_\Omega \frac{1}{2\theta}(u_i - v_i)^2 + |\nabla u_i| \; d\mathbf{x}, \tag{2.27}$$

and a minimization problem for $\mathbf{v}$ with fixed $\mathbf{u}$

$$\min_{\mathbf{v}} \int_\Omega \lambda \; \psi(\rho(\mathbf{v})) + \frac{1}{2\theta}|\mathbf{u} - \mathbf{v}|^2 \; d\mathbf{x}, \tag{2.28}$$

Eq. (2.27) is also known as a total variation image denoising problem using the Rudin-Osher-Fatemi energy [ROF92]. For each component of the variable $\mathbf{v}$ it strives to find a differentiable, scalar field u_i that is on the one hand close to the fixed field v_i but has on the other hand small total variation. Recently, Chambolle has introduced a very elegant, quickly computable and globally convergent solution to this problem, which we will also employ in our minimization framework [Cha04]. He showed, that the solution of Eq. (2.27) is given by

$$u_i = v_i - \theta \; \mathrm{div}\mathbf{p} \tag{2.29}$$

where $\mathbf{p} \in \mathbb{R}^2$ satisfies the partial differential equation

$$\nabla(\theta \; \mathrm{div}\mathbf{p} - v_i) = |\nabla(\theta \; \mathrm{div}\mathbf{p} - v_i)|\mathbf{p}, \tag{2.30}$$

which can be solved by an iterative fixpoint scheme starting with $\mathbf{p} = (0, 0)$ and updating

$$\mathbf{p}^{k+1} = \frac{\mathbf{p}^k + \tau\nabla(\mathrm{div}\mathbf{p}^k - v_i/\theta)}{1 + \tau|\nabla(\mathrm{div}\mathbf{p}^k - v_i/\theta)|} \tag{2.31}$$

with a time step $\tau \leq \frac{1}{8}$.

A remarkable feature of Eq. (2.28) is, that it can be solved pointwise, i.e.

$$\min_{\mathbf{v}} \lambda\psi(\rho(\mathbf{v})) + \frac{1}{2\theta}|\mathbf{u} - \mathbf{v}|^2, \tag{2.32}$$

Information about the values of the function at neighboring points is encoded in the smooth field $\mathbf{u}$ and the gradient is no longer considered in this equation. The liberty to solve only a pointwise problem gives rise to different solution schemes and data-terms in the optical flow literature [ZPB07, SPC09, WPB+09, LLM10] which we will discuss more thoroughly in section 3.3.1. Here we use the alternate exposure image formation model and its differentiated version as data-term $\rho(\mathbf{v})$ in Eq. (2.32). In the next section we show in more detail, how we employ the minimization scheme in our framework.

2.6.4 Implementation

In our case, we employ some small modifications adapted to our problem of minimizing the energy in terms of $\mathbf{w}_1$, $\mathbf{w}_2$ and s. First, we employ the above scheme, i.e., iterating between Eq. (2.27) and Eq. (2.32), by considering $\mathbf{u} = \mathbf{w}_1$, $\mathbf{u} = \mathbf{w}_2$ or $\mathbf{u} = s$, respectively, to solve for each of the unknowns given a fixed approximation of the others. As the thresholding scheme of Ref. [ZPB07] is not directly applicable to our non-linear data-term we apply a descent scheme for Eq. (2.32), profiting from the substitution of the absolute value for the data-term with the differentiable function ϕ_ϵ.

In order to speed up convergence, we implemented the algorithm on a scale pyramid of factor 0.5 (see Sect. 2.5.3), initializing with $s = 0.5$ for the occlusion timing, and $\mathbf{w}_1, \mathbf{w}_2 = 0$ on the coarsest level. On each level of the pyramid we perform several warping iterations where in each iteration we solve for s, $\mathbf{w}_1$ and $\mathbf{w}_2$. For each variable an instance of Eq. (2.27) and Eq. (2.32) has to be solved, Fig. 2.16.

For Eq. (2.27), we employ the dual formulation detailed in Eq. (2.29) and Eq. (2.31) using 5 iterations and a time step of $\tau = \frac{1}{8}$. For all experimental results with the total variation algorithm we use a 5-level image pyramid, 10 warping iterations and 10 iterations to solve Eq. (2.32) and Eq. (2.27). Suitable values for the parameter α, β, γ and θ were found experimentally. For normalized

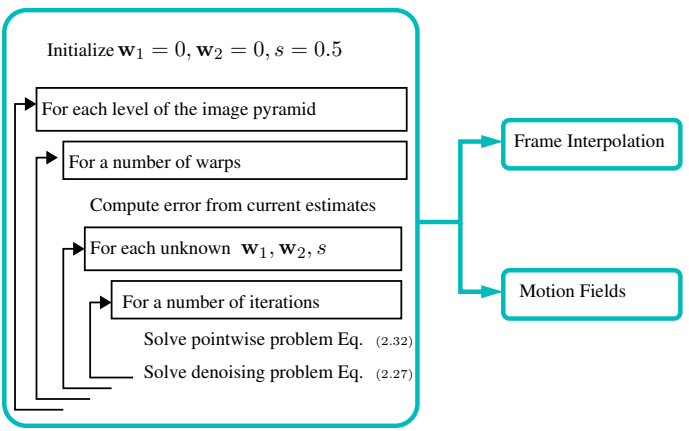


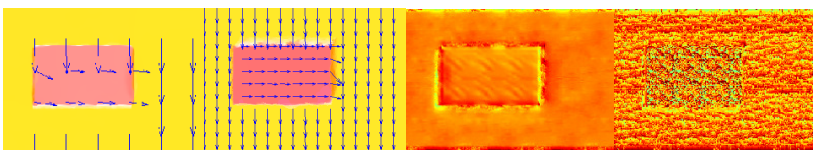
Figure 2.16: The workflow of the total variation approach determines forward and backward motion paths and occlusion times iteratively.

intensity values we found $\theta \in (0, 1]$,$\alpha, \beta \in (0, 0.1]$ and $\gamma \in [0, 0.5]$ to be suitable value ranges. An evaluation of the sensitivity of the algorithm on the parameter choice is given in the next section.

2.6.5 Experiments

The algorithm presented above is dependent on the parameters α, β, γ that steer the weight of the corresponding assumptions and the parameter θ introduced by the optimization scheme. In this section the influence of the parameters is evaluated before more test scenes and a comparison to related algorithms is presented in Sect. 2.7. Working on the 320×250 pixel test scene *square*, Fig. 2.11, the computation time of 191 seconds on a 3.06 GHz processor is independent of the parameters.

The parameters α and θ are equivalent to the parameter $\frac{1}{\lambda}$ and θ introduced by Zach et al. [ZPB07]. As a more detailed evaluation of the aforementioned algorithm by Wedel et al. is available [WPZ+09], we do not discuss the influence



(a) (b) (c) (d)

Figure 2.17: For the synthetic scene *square* shown in Fig. 2.11, subfigure (a) shows the motion field estimated without the color constancy assumption. (b) shows the motion field obtained by using the motion-blur constraint plus color constancy assumption. (c) Regularized occlusion timings and (d) pointwise optimized occlusion timings are hard to evaluate visually, as arbitrary values are allowed for non-occluded points and explicit occlusion segmentation is not performed. Frame interpolation, Fig. 2.18, is better suited for visual evaluation.

of these parameters here.

The parameter $\gamma > 0$ steers the weight of the color constancy assumption, Eq. (2.22), as an additional data-term. Setting $\gamma = 0$ results in visually convincing motion fields, Fig. 2.17(a), where boundaries are well defined. Incorporating the color constancy assumption which formally only holds for non-occluded points decreases the quality of the motion fields at occlusion boundaries, Fig. 2.17(b). However, the numerical error measures for flow fields such as the angular error and the endpoint error are decreased by the color constancy: for the *square* scene the average angular error is 1.7° for $\gamma = 0.2$ and 2.55° for $\gamma = 0$ while keeping all other parameters fixed. Obviously, the motion-blurred data-term, Eq. (2.21), yields a basically correct motion field by integrating along the motion path while the color constancy assumption further increases the subpixel accuracy, Sect. 2.5.2. The actual choice of γ has been found to have only a small influence on the average angular error, Fig. 2.19(a).

The parameter β weights the smoothness of the occlusion timings in Eq. (2.24). For comparison, we estimated motion fields with only pointwise evaluation of the occlusion timings, i.e. $\beta = 0$. Fig. 2.17(c) and Fig. 2.17(d) show examples of the estimated occlusions timings color-coded with the color bar in Fig. 2.12(b).

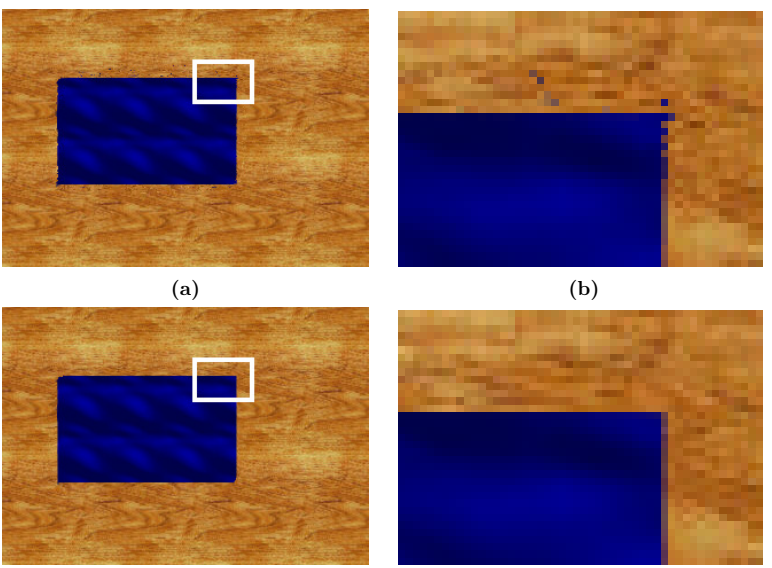
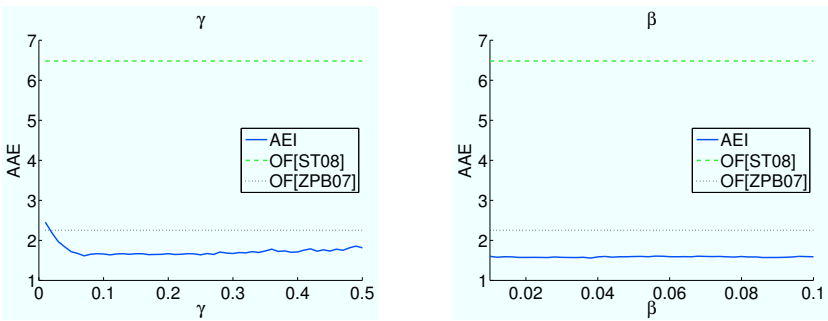


Figure 2.18: (a) Interpolated image at time $t = 0.25$ without regularization: some boundary pixels are wrongly interpolated. (b) Enlargement of the white box in (a). (c) Interpolated image at time $t = 0.25$ with regularization: the interpolation of boundary pixels is improved. (d) Enlargement of the white box in (c).

Appropriate values for s at occluded points are hard to spot in the arbitrary variation of s for non-occluded points. In contrast, erroneous occlusion timings in regions where occlusion actually occurs are easily visible in interpolated images, Fig. 2.18. The interpolated image for $t = 0.25$ using the results of only the pointwise evaluation of the *square* scene is shown in Fig. 2.18(a) or enlarged in Fig. 2.18(b): some pixels at the occlusion boundaries are assigned wrong timings, resulting in foreground pixels when they should show background pixels. The interpolation with regularized occlusion timings, shown in Fig. 2.18(c) or enlarged in Fig. 2.18(d), has hardly any outliers. Apart from these visual artifacts, the average angular error of the motion field with only pointwise occlusion evaluation



(a) (b)

Figure 2.19: Comparison of the average angular error (AAE) for different parameter choices in our algorithm (AEI) to the residual error of state-of-the-art, parameter-optimized optical flow algorithms [ZPB07, ST08]. Keeping all other parameter values fixed, the actual choice of the parameter γ or β has little influence on the AAE: the AAE of our algorithm (blue, solid) is better than for the results of the optical flow algorithms (green, dashed and black, dotted) on a wide interval.

is increased from 1.7° to 2.05°. The assumption of correlated occlusion timings obviously encourages the algorithm to find more consistent motion fields which is advantageous in most real-world scenarios. The actual value of $\beta > 0$ has hardly any influence on overall performance, Fig. 2.19(b).

2.6.6 Discussion

The approach presented in this section is able to estimate dense forward and backward motion paths and dense occlusion timings from a set of three alternate exposure images exploiting the motion similarity of neighboring pixels. Occlusion time estimation is incorporated into the optimization process, so that a separate occlusion detection step is not longer necessary. Optimization is performed by alternating pointwise update and dual smoothing steps using total variation regularization. While a desirable fill-in effect of motion occurs, over-smoothing is

prevented. Although visual evaluation of the occlusion timings is hard due to the indetermined values at unoccluded pixels, frame interpolation shows exact object boundaries. Additionally, the algorithm is quite robust to changing the weights of the different assumptions, so that an application to real-world scenes, where ground-truth motion is not known, seems promising. In comparison to the non-linear least squares optimization, Sect. 2.5, the simplified update scheme of the global optimization and the abandonment of pixelwise computations in MATLAB speeds up computations considerably without any optimization structures.

2.7 Comparison of Different Motion Estimation Algorithms

To evaluate motion field estimation with alternate exposure imaging we consider synthetic test data as well as real-world recordings. For synthetic scenes with known ground-truth motion fields we estimate motion fields with our algorithms as well as with related approaches [ZPB07, ST08, LAG05] and compare the average angular error (AAE) and the average endpoint error (AEE). Additionally, we interpolate intermediate frames using estimated motion paths and occlusion timings and compare them to ground-truth images and images interpolated with ground-truth motion. Note that we cannot evaluate our method on standard test data, like e.g., [BSL$^+$07, BFB94, MNCG01], because these test sets do not provide any motion-blurred images I_B.

We also show results for real-world recordings. The recordings were made with a PointGrey Flea2 camera that is able to acquire short- and long-exposure images alternatingly.

For visualization of the motion fields we use the color map in Fig. 2.12(a) together with a sparse overlay of the motion vectors which are scaled for better visibility. We use image interpolation with our image formation model, Eq. (2.12), as way to evaluate the estimated occlusion timings. This makes the explicit and implicit occlusion detection comparable and especially the latter case easier to evaluate visually.

2.7.1 Motion Fields for Synthetic Test Scenes

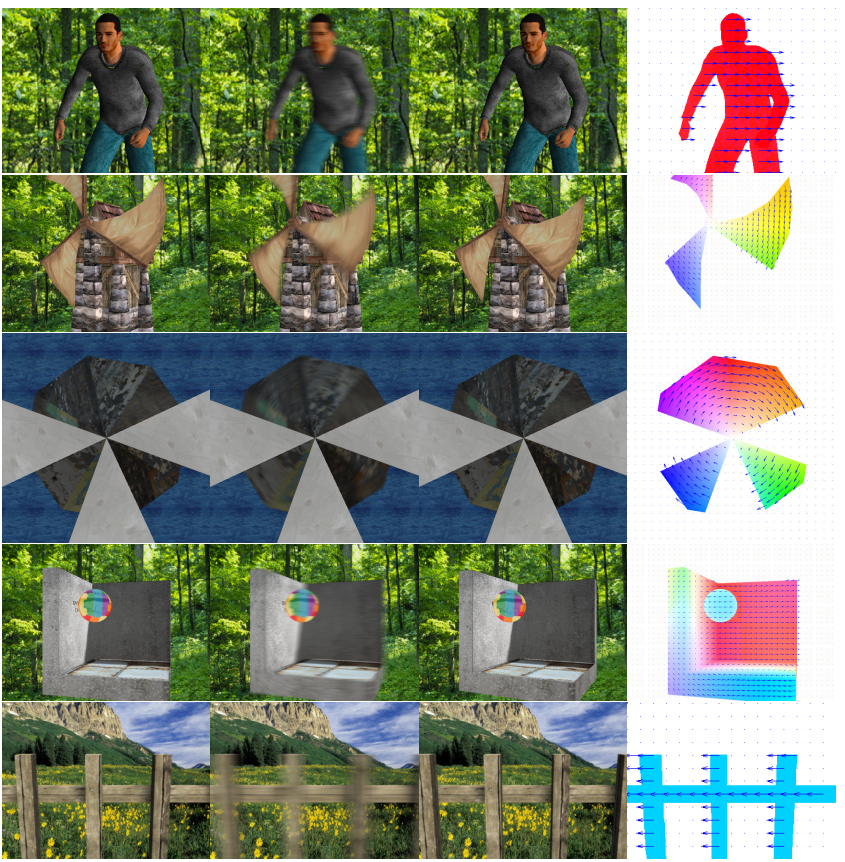
We consider synthetic test scenes containing different kinds of motion. The scene *square*, already introduced in Sect. 2.5.5 for the preliminary evaluation of the optimization approaches, Fig. 2.11, combines 10 pixels per time unit horizontal translational motion of the square with 15 pixels per time unit vertical motion of the background on a 225×320 pixels image, i.e., blurred pixels at the edges of the

square combine several background points as well as several foreground points. The scene *ben*, Fig. 2.20, first row, contains only translational motion in front of a static background. The main challenge of this scene is the large magnitude of the motion, 14 pixels per time unit on a 300×380 pixels image, and background occlusion/disocclusion. The scene *windmill*, Fig. 2.20, second row, contains 7° per time unit rotational motion parallel to the image plane in front of a static background on 800×600 pixels images. In the 512×512 pixels images of the *wheel* scene, Fig. 2.20, third row, the wheel in the background is rotating 7° per time unit while the foreground remains static. The challenge of the 800×600 pixels images in the scene *corner*, Fig. 2.20, fourth row, is out-of-plane rotation of 10° around an axis parallel to the vertical image dimension, while the 320×240 pixels images of the scene *fence*, Fig. 2.20, fifth row, contain translational motion of the same extent as the moving object's width.

To obtain the motion-blurred image I_B we render and average $220 - 500$ images. The first and the last rendered image represent the short-exposure images I_1 and I_2. Ground-truth 2D motion is determined from the known 3D scene motion.

First of all, we test our pointwise least squares approach from Sect. 2.5 and the total variation approach from Sect. 2.6 on the synthetic datasets. We compare the results to state-of-the-art optical flow algorithms, [ZPB07, ST08, LAG05]. For fair comparison, we provide the competing optical flow algorithms also with the short-exposure image $I_{1.5}$, depicting the scene half way between I_1 and I_2. We estimate the motion fields between I_1 and $I_{1.5}$ as well as between $I_{1.5}$ and I_2. The two results are then concatenated before comparing them to the ground-truth displacement field. As optical flow works best for small displacements [LAG05], the error of the concatenation is considerably smaller than estimating the motion field between I_1 and I_2 directly.

We choose the algorithm of Zach et al. [ZPB07], because it relies on the same mathematical framework as our total variation approach. However, our method



(a) I_1 **(b)** I_B **(c)** I_2 **(d)** $\mathbf{w}_{gt}$

Figure 2.20: Together with the scene *square* in Fig. 2.11, the synthetic scenes *ben*, *windmill*, *wheel*, *corner* and *fence* are used to compare alternate exposure and optical flow algorithms. Between (a) the first short-exposure image I_1 and (c) the second short-exposure image I_2 $200 - 500$ images are rendered and averaged to generate (b) the motion-blurred image I_B. (d) The displacement field $\mathbf{w}_{gt}$ is known for every pixel.

uses a long-exposure image instead of a higher frame rate of short-exposure images. We also compare to the algorithm of Sand and Teller [ST08] on three images, because both our methods and their approach consider occlusion effects while estimating motion.

As our algorithms are based on signal-theoretical ideas to prevent temporal aliasing, we incorporate a comparison to the algorithm of Lim et al. [LAG05] that requires high-speed recordings as input. We simulate the high-speed camera with intermediate images such that motion between two frames is smaller than 1 pixel.

Tab. 2.3 shows that our total variation algorithm has the smallest average angular error (AAE) for all test scenes. Also, in all test scenes, except for the rotational motion parallel to the image plane of the scenes *windmill* and *wheel*, our total variation algorithm has the smallest average endpoint error (AEE), Tab. 2.4. The rotation within the image plane directly violates the assumption of linear motion paths in our image formation model, so here our algorithm is outperformed by the TV-L^1 optical flow which does not model the motion paths in the intermediate time between the frames. However, in the *corner* scene with out-of-plane rotation and severe self-occlusion, our total variation algorithm is able to produce the most accurate motion fields in average angular error as well as in average endpoint error.

The least squares approach shows a higher numerical error than our total variation approach in all test cases. Although not competitive to the highly accurate approach of Zach et al. [ZPB07] the least squares approach outperforms the anti-aliased approach of Lim et al. [LAG05] in all but the *fence* scene. In the *fence* scene the least squares approach fails to assign correct motion to the large occluded areas, as nearly all moving points in the image are occluded or disoccluded between I_1 and I_2. For the test scenes with planar motion, the least squares algorithm achieves results competitive to the occlusion aware optical flow algorithm of Sand and Teller [ST08], while the motion field for the out-of-plane rotation of the *corner* scene is less accurate.

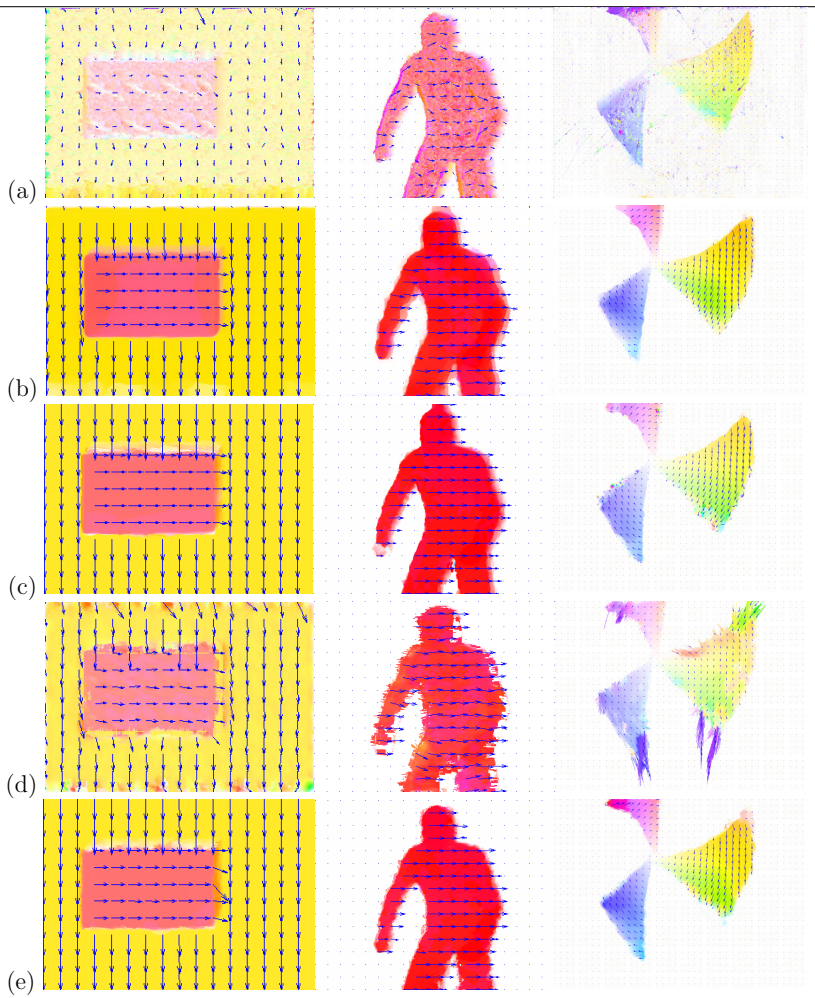


Figure 2.21: For comparison, motion fields for the datasets *square*, *Ben* and *windmill* are calculated with different algorithms. (a) The algorithm of Lim et al. [LAG05] requires a high number of input images and returns noisy motion fields. (b) While the approach of Sand and Teller [ST08] is prone to over-smoothing, (c) the approach of Zach et al. assigns unpredictable motion fields to occluded points. (d) Spurious assignments at occlusion boundaries and insufficient regularization in textureless regions deteriorate the quality of our least squares approach. (e) Our total variation approach to alternate exposure imaging consistently shows the most accurate motion fields of all approaches.

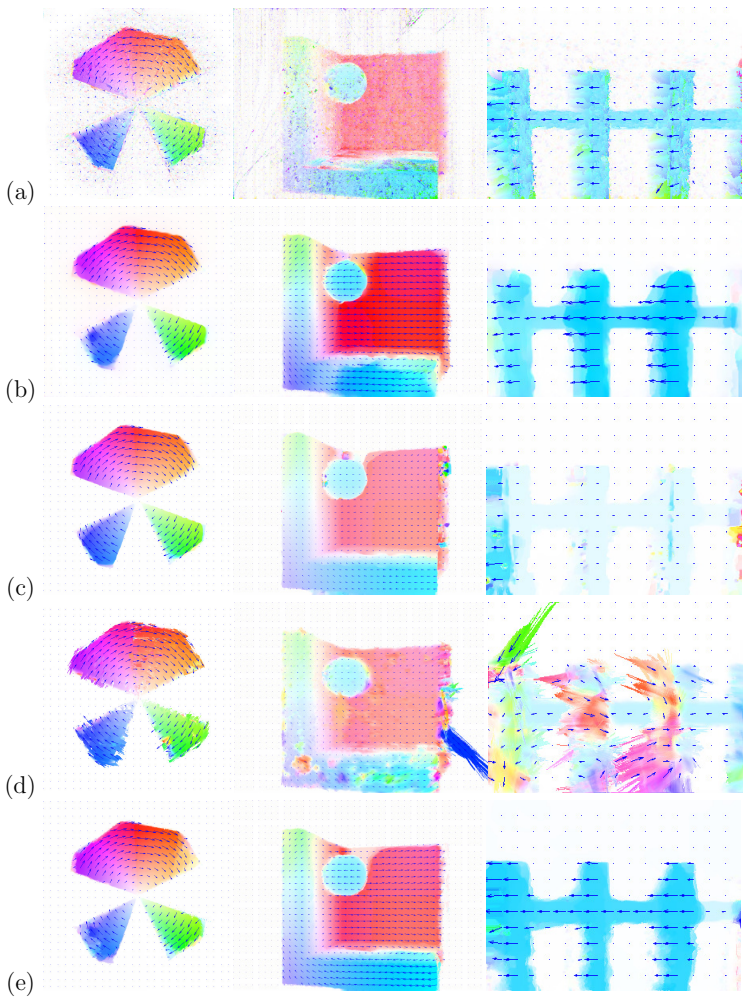


Figure 2.21: (cont.) Motion fields for the datasets *wheel*, *corner* and *fence* are calculated with different algorithms.

AAE [°]	*ben*	*square*	*windmill*	*wheel*	*corner*	*fence*
Sand, Teller [ST08]	8.42	6.48	6.78	13.39	6.40	19.12
Zach et al. [ZPB07]	5.81	2.25	4.87	2.59	5.05	19.44
Lim et al. [LAG05]	9.01	12.19	49.63	27.29	38.40	34.17
AEI, least squares	6.31	6.24	8.64	4.19	12.87	34.41
AEI, total variation	**4.27**	**1.70**	**4.56**	**2.21**	**4.57**	**12.97**

Table 2.3: Comparison of different motion estimation methods for six synthetic test scenes: the motion fields computed using the total variation approach to alternate exposure imaging (AEI) consistently yields a smaller average angular error (AAE) than the least squares approach and competitive optical flow algorithms given three images [ZPB07, ST08] or sequences of temporally oversampled images [LAG05].

AEE [px]	*ben*	*square*	*windmill*	*wheel*	*corner*	*fence*
Sand, Teller [ST08]	0.91	5.72	2.95	1.27	2.85	3.36
Zach et al. [ZPB07]	0.59	0.62	**1.69**	**0.60**	1.27	14.75
Lim et al. [LAG05]	1.46	4.88	7.69	1.82	7.73	5.23
AEI, least squares	0.99	1.73	5.47	1.02	6.30	12.64
AEI, total variation	**0.57**	**0.52**	2.16	0.61	**0.92**	**2.62**

Table 2.4: For the six synthetic test scenes, the average endpoint error (AEE) of the total variation approach to alternate exposure imaging is among the smallest in comparison to competitive optical flow estimation algorithms given three images [ZPB07, ST08] or sequences of temporally oversampled images [LAG05].

Visual comparison of the motion fields, Fig. 2.21, shows, that the small numerical error of our total variation approach is due to several reasons: While the algorithm of Lim et al. [LAG05] returns noisy motion fields, Fig. 2.21(a), the algorithm of Sand and Teller [ST08] tends to over-smooth motion discontinuities, Fig. 2.21(b). The TV-L^1 optical flow algorithm [ZPB07] assigns outlier motion vectors to occluded points, Fig. 2.21(c). The quality of the least squares alternate exposure algorithm suffers considerably from noisy motion path detection and spurious motion assignments at non-detected occluded points, Fig. 2.21(d). In contrast, the total variation approach to alternate exposure imaging stands out due to sharp motion boundaries and appropriate motion assignment at occlusion borders, Fig. 2.21(e).

2.7.2 Frame Interpolation for Synthetic Test Scenes

For evaluation of the occlusion time estimation, we interpolate intermediate images based on Eq. (2.12). Frame interpolation makes the explicit and the implicit occlusion detection in the least squares and the total variation approach comparable: In the least squares approach occlusion is only determined where the optimization residual exceeds the threshold T_E. In contrast, the total variation approach estimates occlusion timings for all points, although they are well defined only where occlusion actually takes place, Fig. 2.22. Frame interpolation allows to evaluate the occlusion timings objectively. For comparison we also interpolate intermediate frames between I_1 and $I_{1.5}$ and between $I_{1.5}$ and I_2 using the method introduced by Baker et al. [BSL$^+$07] and using blending of forward and backward warped images. None of the two methods considers occlusion. We compare the results of each interpolation method and the results of the native frame interpolation from the alternate exposure image formation, Sect. 2.4.4, to the ground-truth intermediate images. Fig. 2.23 gives an overview of the sum of squared differences (SSD) for all test scenes.

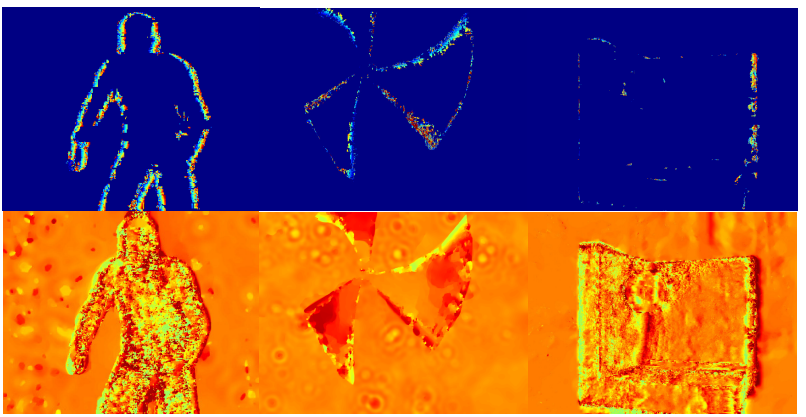


Figure 2.22: Occlusion timings of the least squares approach are determined only where the optimization residual exceeds a threshold (top row). With the total variation approach occlusion timings are determined for every pixel, but are only well-defined at occlusion boundaries (bottom row). Easier comparison of occlusion timings can be obtained by considering frame interpolation, Fig. 2.24

Although the least squares algorithm has a higher AAE/AEE than the optical flow algorithm of Zach et al.[ZPB07] the interpolation error for some of the images is considerably smaller than using the optical flow algorithm with any of the two interpolation methods. The interpolation with the motion paths from the total variation approach consistently shows better interpolation results than the optical flow based interpolation. Especially, both the least squares and the total variation algorithm occasionally obtain a smaller SSD than interpolation with ground-truth motion. This is due to the fact that inaccuracies in the motion fields can be balanced by the successful handling of occlusion boundaries, as can also be seen in Figs. 2.24 and 2.25, where images interpolated at $t = 0.25$ using our algorithms are compared to the corresponding images interpolated with the method of Ref. [BSL$^+$07] and forward/ backward warping using ground-truth motion.

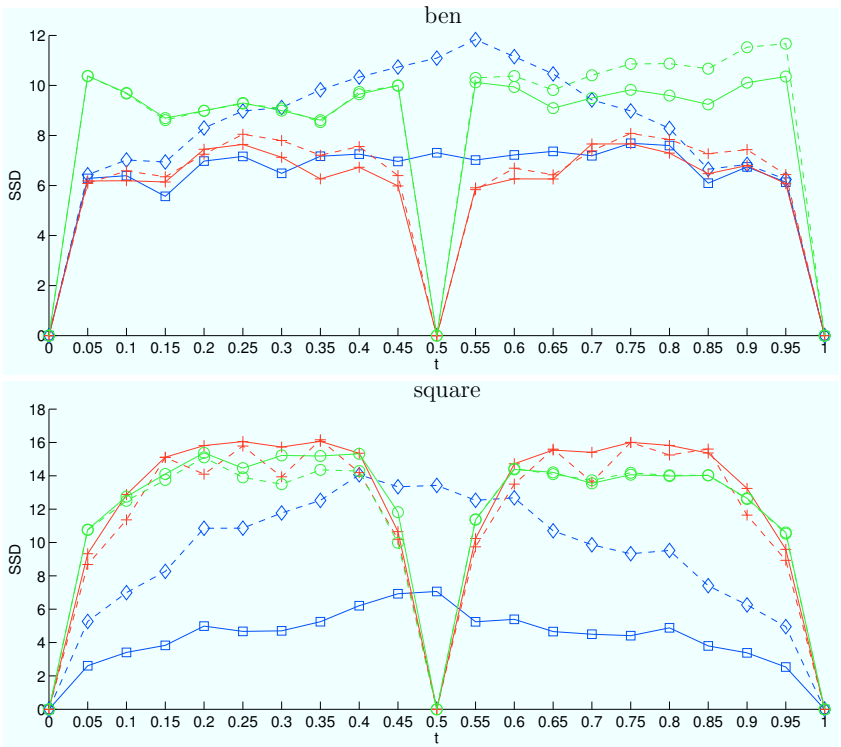


Figure 2.23: The sum of squared differences (SSD) between interpolated images and ground-truth images. The dashed green (circled) line shows the SSD for forward interpolation with optical flow [ZPB07], while the continuous green (circled) line shows the SSD for forward-backward interpolation using the same optical flow. Red (crossed) dashed and continuous lines indicate the SSD for forward interpolation [BSL$^+$07] or forward-backward interpolation, respectively, using ground-truth motion fields. The SSD obtained using least squares optimization for motion paths from alternate exposure imaging is indicated by the blue dashed line (diamonds) and the SSD obtained using total variation regularization for the motion paths is indicated by the blue continuous line (squares).

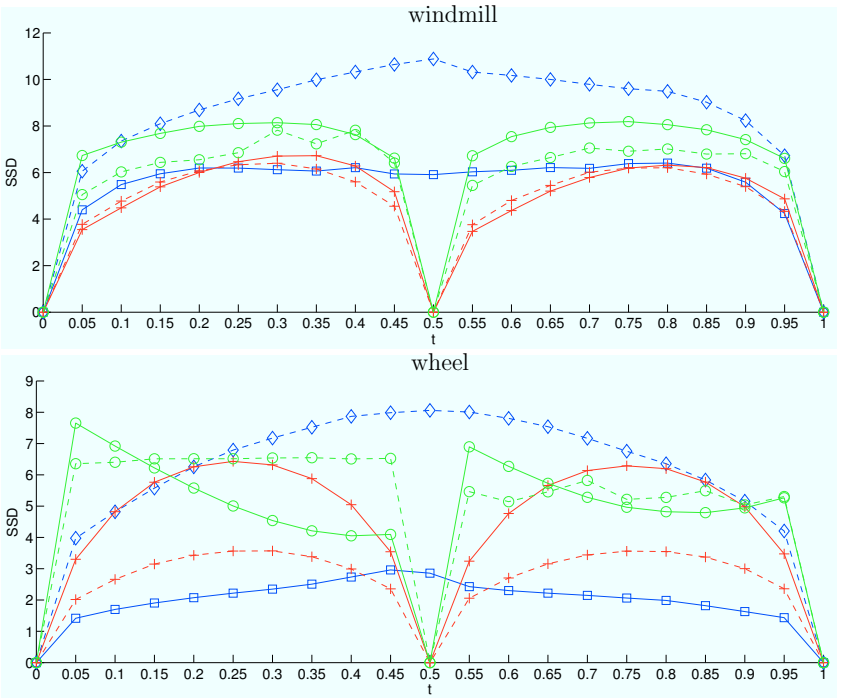


Figure 2.23: (cont.) The sum of squared differences (SSD) between interpolated images and ground-truth images using different interpolation algorithms and different motion estimates.

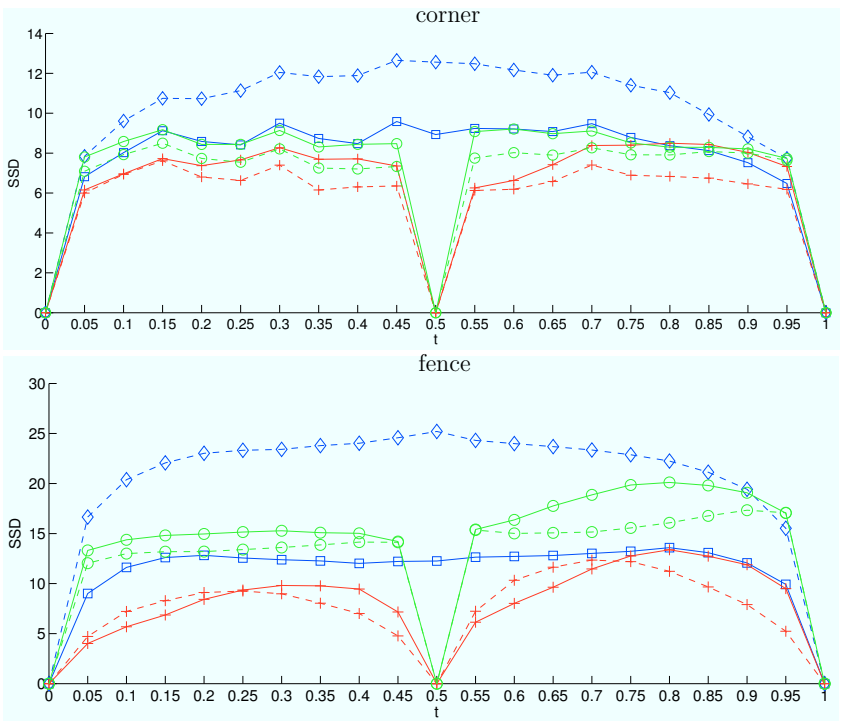


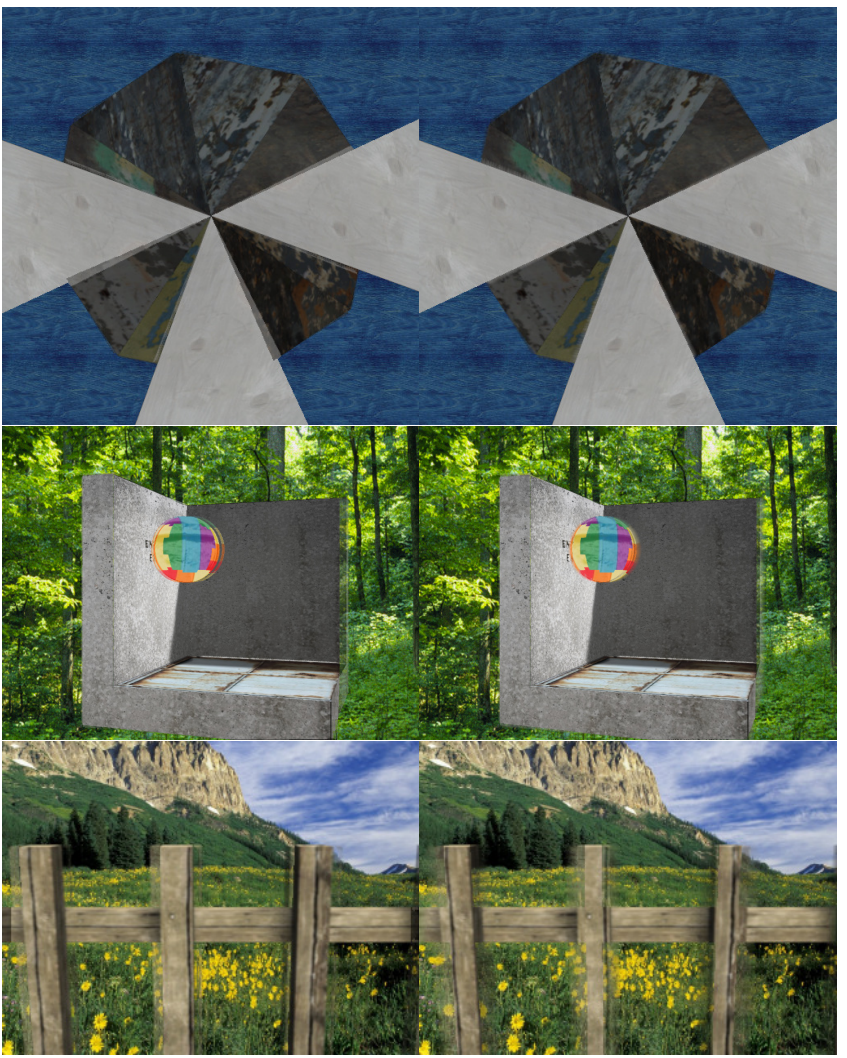
Figure 2.23: (cont.) The sum of squared differences (SSD) between interpolated images and ground-truth images using different interpolation algorithms and different motion estimates.

(a) **(b)**

Figure 2.24: (a) Interpolation at $t = 0.25$ with the method proposed in [BSL+07] and (b) blending of forward- and backward-warped images show artifacts at occlusion boundaries even when ground-truth motion fields are used, because occlusion information is not available (Scenes *square*, *ben* and *windmill*).

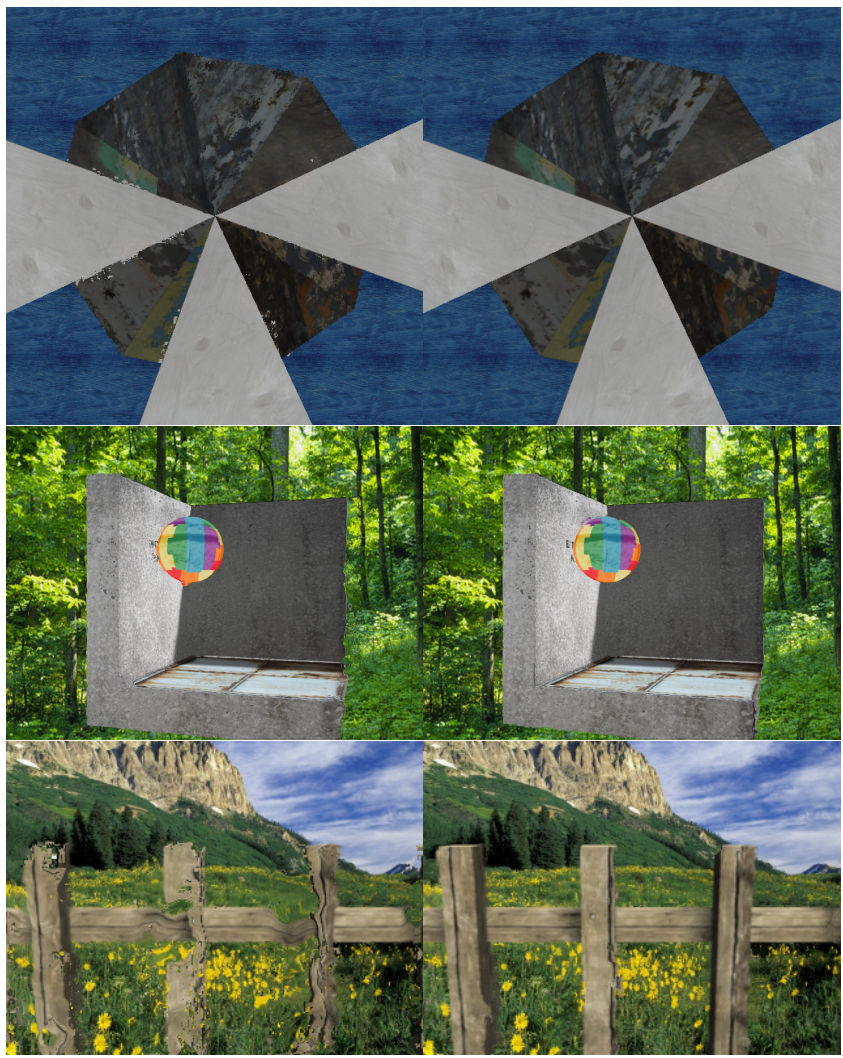
(c) **(d)**

Figure 2.24: (cont.) (c) Thresholded occlusion detection in the least squares approach to alternate exposure imaging fails to detect occlusion at some boundaries and exhibits remaining artifacts. (d) Interpolation with total variation regularized motion paths and occlusion timings reduces artifacts at occlusion boundaries (Scenes *square*, *ben* and *windmill*).



(a) **(b)**

Figure 2.25: (cf. Fig. 2.24) Frame interpolation with ground-truth based methods for the scenes *wheel*, *corner* and *fence*.



(c) (d)

Figure 2.25: (cont.) Frame interpolation with alternate exposure methods for the scenes *wheel*, *corner* and *fence*

2.7.3 Real-World Recordings

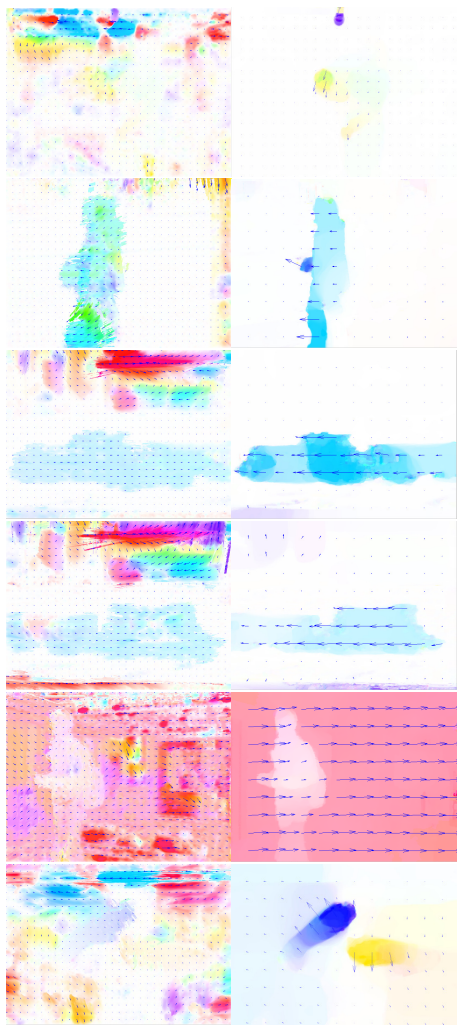
We also test our methods on real-world recordings. We use the built-in HDR mode of a PointGrey Flea2 camera to alter exposure time and gain between successive frames. By adjusting the gain, we ensure that corresponding pixels of static regions in the short-exposure and long-exposure images are approximately of the same intensity. With the HDR mode we are able to acquire I_1, I_B and I_2 with a minimal time gap between the images. The remaining gap is due to the fixed 30 fps camera frame rate and the readout time of the sensor. As for the synthetic test scenes, we also record a number of real test scenes with different challenges. Thereby, all images are recorded with the same PointGrey Flea2 camera and a resolution of 480×640 pixels. The scene *juggling*, Fig. 2.26 first row, contains large motion of a small ball, that additionally vanishes from the field of view of the camera. To ensure that the short-exposure images contain no or only little motion-blur, their exposure time is set to 6.02 ms. However, the camera can only process an image every 33.33 ms. Using only short-exposure images, this would lead to 27.31 ms of unrecorded motion between sharp images. For our method, we record a long-exposure image with an exposure time of 39.65 ms. With our camera setup we measured a remaining gap between I_B and the succeeding short-exposure image of 0.48 ms which is due to readout time of the sensor and other hardware constraints. I_B reduces the gap and provides us with temporally anti-aliased information.

The same camera setting was used for the *walking* scene, Fig.2.26, second row, where a person walks by on a street and the leg moves in the order of magnitude of its width. The scenes *model train 1* and *2*, Fig.2.26, third and fourth row, are recorded with the same camera setting. Challenges in these scenes are the moving shadows and the highlight on the wagons that violate the assumption that motion is the only reason for brightness changes in the scene. To test the flexibility of the approach to different foreground and background motions, the scene *tracking*, Fig.2.26, fifth row, was recorded with a camera following the motion of the person

in the foreground, i.e., objects in the background have a relative motion to the camera according to their depth. For the *waving scene*, Fig.2.26, sixth row, we use exposure times of 20.71 ms and 124.27 ms, resulting in measured gaps of 12.45 ms and 0.48 ms, respectively. This scene provides different motions, i.e., that of the hands moving in opposite direction and the static background and the occluded texture of the eye.

The recorded images and the motion fields estimated with the least squares and the total variation approach are shown in Fig. 2.26. While motion fields estimated by the least squares approach are mainly dominated by noise, closer inspection shows that in places where motion actually occurs it is often detected correctly, for example the ball flying out of the image in the *juggling* scene. Only the large, sparsely textured regions in the background do not provide enough information for the pointwise approach, so that *any* noise in the image is able to produce pronounced incorrect motion estimates. The results of the total variation approach look more promising. Although the background often provides only little texture, motion is generally estimated correctly. In the *juggling* scene one of the balls still visible in I_1 disappears in I_2, making standard optical flow computations infeasible, but the motion-blurred image captures the path taken by the ball and enables correct motion field estimation. For this reason, our methods can even handle the small ball leaving the picture. In the *walking* scene, the total variation approach is not only able to detect the motion of the leg moving approximately as far as its width, but also the motion of the hand faithfully. Despite the similar color of the jacket in arm and body part, the motion boundary is detected. In the scenes *model train 1* and *2* the total variation approach shows robustness to moving shadows and the highlights on the last wagon. In the *tracking* scene both algorithms detect the motion of the dark backpack in front of the the dark background correctly, and the total variation algorithm additionally is able to faithfully detect the motion of both hands. Also for the *waving* scene, the total variation algorithm is able to cope with the motion and the occluded texture.

Figure 2.26: The built-in HDR mode of PointGrey cameras is able to alter exposure time and gain between succeeding frames so that (a) short, (b) long, and (c) short exposures can be successively acquired at comparable brightness and with minimal temporal gap between frames.



(d) (e)

Figure 2.26: (cont.) Motion fields for the real-world scenes (from top to bottom) *juggling*, *walking*, *model train 1*, *model train 2*, *tracking* and *waving* are estimated with (d) the least squares approach, Sect. 2.5 and with (e) the total variation approach, Sect. 2.6.

2.7.4 Evaluation

The results of the experiments require the discussion of two issues. First, we discuss the advantages and disadvantages of the two approaches to alternate exposure motion estimation, i.e., of the least squares approach presented in Sect. 2.5 and the total variation approach presented in Sect. 2.6. Then we compare both approaches to optical flow approaches that consider only short-exposure images.

Comparison of the Two Alternate Exposure Approaches

The least squares approach to optical flow from alternate exposure imaging shows that the image formation model introduced in Sect. 2.4 is sufficient to allow for motion estimation. In some of the synthetic scenes it outperforms modern optical flow algorithms [ST08, LAG05] that are designed to handle occlusion or deal with temporal aliasing. As no regularization is necessary and the approach solves ambiguities by additional assumptions, the resulting motion fields seem visually quite noisy. For real-world recordings, the algorithm is very susceptible to noise and inaccuracies in the gain correction of long- and short-exposure images due to its squared error term and the requirement to solve the equations for all occlusion timings simultaneously.

The total variation approach requires regularization to solve the ambiguities of the image formation model for unoccluded points. Thus, the estimated motion fields look visually more pleasing, and the evaluation for synthetic scenes shows that they are indeed more accurate than comparable state-of-the art optical flow algorithms [ZPB07, ST08, LAG05]. Due to implicit occlusion handling, the total variation approach can also deal with objects where every moving pixel is an occluding pixel - a situation like in the *fence* scene where the least squares approach fails. The images interpolated using the motion paths of the total variation approach have also more exact occlusion borders than using the least squares approach, where undetected occlusion borders occasionally corrupt the interpolation. Finally, the total variation approach estimates convincing motion fields also

for real-world recordings even for only roughly adjusted brightness levels between the short-exposure and long-exposure images.

Limitations and Advantages from Alternate Exposure Imaging in Image Based Motion Estimation

Motion field estimation from alternate exposure imaging shares some of the limitations inherent to all optical flow methods. Like in all purely image-based methods, motion in poorly textured regions cannot be detected robustly. This can be seen in the black background of the *waving* scene, Fig. 2.26.

Also common to all optical flow methods, we assume that motion is the only source of change in brightness, disregarding highly reflecting and transparent surfaces from the calculations.

Furthermore, we made the assumption that the short-exposure images are free of motion blur. Practically this is true if motion during the short exposure time is smaller than half a pixel.

Image noise is also a common problem in motion estimation. In the short-exposure images of the alternate exposure images, the gain and therefore the noise level is increased in comparison to optimal short-exposure images. While the least squares approach is indeed susceptible to noise, the use of a suitable penalizer for the data-term and the total variation regularization deals with noise successfully. Additionally, for non-occluded points the total variation algorithm can choose the occlusion timing s so that noise with zero mean in the path integral can cancel out much better than in the customary comparison of two single pixels.

In contrast to most optical flow methods, we are able to include occlusion explicitly into our image formation model. With the total variation approach arbitrarily large occlusion as well as disocclusions can be handled under the assumption that a scene point changes its state of visibility only once. This assumption on the visibility state infers that, e.g. for a static background point an occluding object can move at most as far as its width before the background point reappears.

Our image formation model works with motion paths instead of displacement fields. While motion paths can theoretically have arbitrary forms, the assumption that they are linear allows for a simple parametrization. Actually, linear motion paths imply that the displacement of all pixels on the path is uniform and of constant speed. But as motion paths are allowed to vary for neighboring pixels, the approach can successfully handle also much more complex motions.

Finally, while recording the alternate exposure sequence, we replace one short-exposure image with a long-exposure image. To show the sequence to a viewer uninterested in motion detection, the long exposed frame may simply be skipped, or, to ensure a sufficient frame rate, intermediate images can be easily and quite faithfully interpolated with the proposed method.

2.8 Summary

In this chapter we have analyzed the problem of temporal aliasing in the traditional optical flow approach. We found that motion-blurred images are correctly pre-filtered for motion estimation, but they lack high image frequencies. As a compromise we introduce alternate exposure imaging to record motion information as well as high frequency content. Based on our image formation model, we propose two algorithms that are able to estimate dense motion fields and also occlusion timings. The first algorithm is able to perform the estimation without any regularization, that is usually necessary to solve the aperture problem in optical flow estimation. Although competitive on synthetic data, the lack of regularization makes the pointwise least squares approach susceptible to image noise and gain-maladjustment in real-world recordings.

In contrast, our total variation approach is not only more accurate than state-of-the-art optical flow on synthetic scenes, but it also shows convincing performance on real-world scenes. Notably, it is able to handle occlusion situations where the state-of-the-art in optical flow - based on two successive images - is

destined to fail. In our experiments, we also observed that accuracy of the motion field is not the most important issue for frame interpolation. With our estimated motion fields, that contain some residual error, together with occlusion timings, we are able to obtain interpolated frames that have a smaller numerical error than interpolation with ground-truth motion. In addition, the interpolated frames also look perceptionally convincing, as, in contrast to traditional interpolation, our algorithm is able to reproduce occlusion borders correctly by making use of the estimated occlusion timings.

While alternate exposure imaging provides an adequate way to obtain dense motion fields and occlusion timings for a video sequence, modified shutter timings and the recording of motion-blurred images work only within a temporal sequence. If, in contrast, dense correspondence fields between two different camera views are required, the approach is not applicable. As situations with several video cameras also provide additional information on a scene that traditional optical flow algorithms do not exploit, we look more closely at this situation in the next chapter.

Chapter 3

Correspondence Estimation for Multi-View Video Setups

3.1 Introduction

In recent years the increased availability of inexpensive video cameras together with cheap storage space and fast data transfer has led to a growing interest in multiple view video [MP04, WJV+05]. Especially the demand for stereoscopic video content has been largely increased by the spread of stereoscopic displays as well as the revival of stereoscopic motion pictures [Lip07]. Dense correspondences between the images of unsynchronized multi-view video sequences allow, e.g., to exploit the space-time manifold spanned by the recorded images [LLB+10] or reconstruct depths of moving objects [ZT03]. Usually, these correspondences are estimated between pairs of images. In the case of multi-view video, this can lead to inconsistencies: processing first the next frame in time and then the next frame in space can end up with a totally different result than first processing the spatially neighboring frame and then the temporally neighboring frame, Fig.3.1. A correspondence estimation algorithm for multi-view video should therefore consider spatially and temporally neighboring frames simultaneously right from the beginning.

Many algorithms considering multiple cameras assume these cameras to be

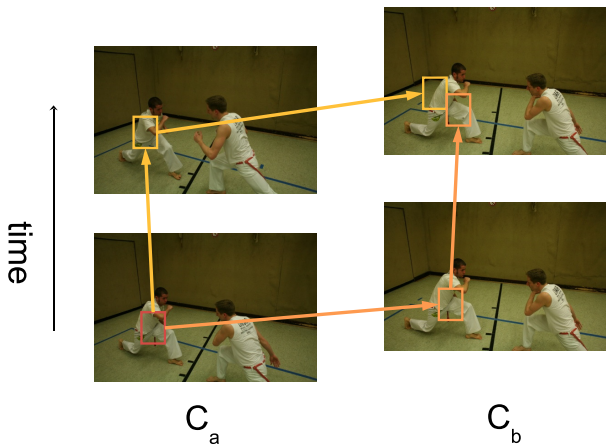


Figure 3.1: Small errors in pairwise estimated correspondence fields accumulate quickly: Going from the lower left image to the upper right first in time and then in space (yellow) yields a different pixel correspondence than first in space, then in time (orange).

synchronized and calibrated. In this case the application of stereo correspondence algorithms with their restricted solution space is possible [SS02]. However, if cameras are moved during recording, e.g. for aesthetic reasons, calibration information is lost. Also, multiple cameras are hard to synchronize in an outdoor environment [TVG04]. While several methods for frame accurate synchronization of cameras have been proposed [TVG04, MSMP08, HRT+09], sub-frame accurate synchronization requires special hardware setups and is usually not possible when using inexpensive consumer cameras. In our approach we therefore estimate general dense correspondence fields, i.e. optical flow, between spatially as well as temporally neighboring images. Synchronization of the video streams in post production can then be performed applying our more accurate and reliable multi-video flow, e.g. using the temporal frame interpolation proposed by

Wang et al. [WSY07].

In our approach we ensure temporal and spatial coherence by considering closed loops of three neighboring images. Additionally to the usual image space constancy, i.e., brightness constancy or feature descriptor constancy, we require correspondences within any set of three neighboring images to be consistent. Considering dense correspondence estimation, the consistency constraint provides us with a parameter-free measurement for the reliability of the flow. Evaluation of this measurement during optical flow estimation helps us to establish more accurate flow fields. In addition, it allows us to detect occluded areas and to suppress the brightness constancy assumption for points where it is not valid.

Consistency between adjacent frames is not only an additional constraint for dense correspondence fields, it is also a constraint for sparse feature matching. Features corresponding to one scene point should be matched consistently in three neighboring images. In contrast to optical flow algorithms, feature matching can handle large displacements easily. Features are matched based on a feature descriptor that is independent from the absolute position in the image. This allows feature matching to deal with arbitrary distances, as long as the features remain recognizable. We show in our work that with the increased robustness provided by the consideration of consistency, feature matches can be directly included into the optical flow framework. This allows us to detect large motions more robustly while maintaining a high accuracy.

In the next section we first review work related to our approaches, before we introduce our loop-consistent dense optical flow algorithm, Sect. 3.3, our loop-consistent feature matching, Sect. 3.4, and the combination of loop-consistent dense flow estimation with feature matching, Sect. 3.5.

3.2 Related Work

Estimating dense correspondence fields for multi-view image sequences is related to several fields of research. First, correspondences can be calculated independently between any pair of images using one of the optical flow methods discussed in Sect. 1.3. The methods introduced there also cover approaches that profit from the temporal coherence within each individual video sequence. Yet, to ensure consistency in multi-view video sequences, correspondence fields in temporal direction and between cameras need to be considered together. Approaches that consider spatially and temporally neighboring images for dense correspondence estimation are discussed in the first subsection. But consistency is not only used during the estimation of dense correspondence fields. It can also be considered as a confidence measure for readily estimated optical flow fields. We therefore give a synopsis of confidence measures in optical flow estimation in the second subsection. The issue of consistency has mainly been raised in the field of sparse feature matching. We give a short overview over consistent feature matching in the third subsection. Related to our work are also techniques to obtain dense correspondence fields from sparse correspondences by intelligent interpolation - exploiting the robustness of feature matching to large motion, noise and changes in illumination also for applications that require dense correspondences. We discuss these approaches in the last subsection of this section.

Consistent Dense Correspondence Estimation

For consistent multi-video editing, correspondences are required between spatially as well as temporally adjacent images. In literature, so far, consistency is considered only for correspondences between synchronized cameras. If the cameras are synchronized or the scene is static, epipolar geometry between images can be exploited. The solution space for correspondences between spatially adjacent images is restricted to the epipolar lines and stereo algorithms can be applied to estimate the disparities [SS02]. If disparities are to be estimated for a sequence of stereo

image pairs, a problem also known as *temporal stereo*, the temporal coherence in the sequence can be used to regularize disparity estimation [LALS04, SW09].

If, in addition to the disparities between the synchronized images, the optical flow within each sequence is estimated, the apparent 3D motion of scene objects, the *scene flow* [VBSK00], can be established. While some approaches to estimate scene flow treat disparity estimation and optical flow estimation between temporally neighboring frames as separate problems [LS08a], other approaches estimate disparity and optical flow in a common variational framework [WRV+08, HD07, MS06]. But these approaches still need the initialization with separately estimated disparities [WRV+08] or two-image optical flow [HD07, MS06]. As the considered energy functionals are in general non-convex, errors in the correspondence field used for initialization can lead to results that differ largely from the desired correspondences.

A further possibility to render scene flow estimation more robust is to check the symmetry of disparities and optical flow between image pairs during estimation [BJK08], leading to symmetric but not necessarily consistent flow fields. Bolles and Woodfill eliminate wrong correspondences from the independent disparity and optical flow estimation by symmetry and spatio-temporal checks [BW93]. As inconsistencies are only inspected *after* correspondence estimation is terminated, this procedure results in non-dense correspondences.

Confidence Measures for Optical Flow Estimation

The scene flow approach of Bolles and Woodfill [BW93] implicitly establishes a confidence measure based on consistency between four images. This measure allows to eliminate outlier estimates in the correspondence field. In our work we concentrate on loops of correspondences between three images. While both, three spatio-temporal neighboring images or two succeeding stereo pairs, provide information in temporal as well as spatial direction, three images are actually the

minimal number of images to check loop-consistency and allow to locate possible errors more precisely.

Generally, confidence measures for correspondence field estimation are valuable to discover uncertainties. If regions of uncertain correspondence estimation are known, applications can decide to reinitialize rather than to follow an uncertain correspondence. Or, methods for motion inpainting can be applied to replace uncertain optical flow estimates [BKGR09]. Confidence measures are of importance for optical flow estimation, as the problem is ill-posed and solutions therefore are often unreliable [BFB94].

A first class of confidence measures are based on the input images alone and consider where optical flow can be estimated in a reliable way [BSL96]. Some of these confidence measures, like the strength of the image gradient [BFB94] or the rank of the linearized system for optical flow estimation [NDB09], can be evaluated even before the flow is estimated and thus save the work of estimating an optical flow that is not used in the end. As a disadvantage, they depend heavily on the optical flow approach to be applied and are very susceptible to noise in the input image. Additionally image based confidence measures do not consider that for a vanishing gradient, the fill-in effect of the smoothness constraint usually provides very good results in spite of the locally undefined motion [BW06].

A second class of confidence measures accounts for the measured image data as well as the estimated flow. These methods consider the optimization residual of the optical flow estimation [BW06]. Yet, by design this approach cannot detect erroneous flow due to the wrongful enforcement of model assumptions, e.g., flow deviations due to occlusions or image noise. Furthermore, they depend on the parameters that steer the weight of the model assumptions such as smoothness constraint and data-term.

A further class of confidence measures evaluates only the estimated flow itself, discarding the measured image from the considerations. In Refs. [KKJG07] and [KMG08] Kondermann et al. consider subspace and statistical models to establish confidence measures. Yet, both proposed confidence measures require

training data of the expected flow to learn the parameters of the respective model. Although the authors propose to use the estimated flow itself as training set, this requires the flow to be correct in average. Thus the approach fails especially in those cases where optical flow estimation is badly flawed.

Consistent Sparse Correspondence Estimation

While validation and consistency checks of dense correspondence fields are considered only by comparatively few authors, validation of sparse image feature matches has received much more attention. Thereby, the term *feature* might refer to edge pixels, corners or blob-like image structures [MTS+05]. In spite of the increasing quality in modern feature detection, description and matching, mismatches of features are common [MS05] and researchers look intensely for ways to exclude wrong matches that go beyond the images' color or gray value distribution.

If features are matched between two images from synchronized cameras or for mostly static scenes, spurious matches can be discarded using epipolar geometry [SZ02, HZ03]. Generally, the assumption of global affine motion between two images can be used to validate matches [BGPS07]. But also game theoretic approaches exploiting local similarity transforms are used to establish reliable matchings between two images [ART10].

If several independent objects move in a monocular sequence, e.g. for person or object tracking [YJS06], feature locations from previous frames can be used to estimate feature locations in the current frame [Zha94]. Assuming that features have at most one correct match in each frame, disjoint tracks of features over multiple frames can be considered to improve matches [VRB03, SS05, SSS06]. Thereby, the tracks provide a regularization of the matches over time, but no feedback for the correctness of the tracking is provided.

For static scenes, the trifocal tensor [TZ97] can be used to consider consistency of the matching between more than two images [BTZ96]. Yao and Cham

first verify and add matches between image pairs to satisfy the epipolar constraint, before the matches are extended to image triples and the trifocal tensor is computed [YC07]. In contrast, Zach et al. first determine global, invertible transformations between image pairs before they detect wrong transformations on multi-image loops and discard them [ZKP10], enabling more robust multi-image 3D reconstruction.

If a dynamic scene is recorded by multiple, unsynchronized cameras, Ho and Pong work with high density feature points and use assignments of neighboring pixel in a relaxation labeling framework to obtain consistent matchings [HP96]. That is, they employ exactly the spatial smoothing that often fails to capture motion details in optical flow estimation. In the same setup, Ferrari et al. perform consistency checks on loops of images, but require an additional similarity measure that is different from the measure used to establish the preliminary matchings [FTV03].

Mathematically, the problem of finding correspondences on three sets of equal, finite cardinality is well studied [Spi00] and approximation algorithms to the NP-hard problem have been proposed by several authors [CS92, BCS94].

Integration of Sparse and Dense Correspondence Estimation

There are three basic approaches to extend sparse matches into dense correspondence fields. All of them pay careful attention to exclude outliers in the process. The first approach is built on dense optical flow estimation. For instance, Brox et al. [BBM09] include sparse correspondence vectors based on the 5 most similar region matches as a prior into their optical flow formulation. They weight the matches with a confidence measure proportional to their descriptor distance. Thus they are able to faithfully detected moving extremities, which are undetectable to standard optical flow due to their ratio of size and motion. Still,

outlier matches decrease the quality of the flow in some parts, e.g. in the background of the scenes. In their recent journal version, Brox and Malik therefore propose symmetric matching between features, before they include the matches into the optical flow estimation [BM10], focusing on the monocular setup. Xu et al. [XJM10] determine SIFT correspondences on each level of the scale pyramid and assign an initialization for the next level based on non-local distinctive SIFT correspondences and the upscaled version of the previous flow estimate. With the considerable computational effort invested in this method, motion detail can be preserved, if a corresponding SIFT-based motion vector is detected somewhere in the image and matched successfully.

The second approach to transform sparse matches into a dense correspondence field is based on region growing. L'Huiller and Quan [LQ02], Strecha et al. [STV08] and Goesele et al. [GSC+07] consider spatial neighborhoods of initially matched features and propagate matches as a function of the matching error of each newly assigned pixel. Especially the latter method is notable. In contrast to the other methods it allows to revisit pixel assignments and actually provides sub-pixel improved correspondence vectors for each of the considered pixels. As drawback, several heuristics have to be applied to prevent oscillation and enforce convergence of the algorithm.

As a third approach, motion segmentation based methods like those of Stich et al. [SLW+10] and Wills et al. [WAB03] first estimate a set of planar homographies based on initially matched features. The former uses color based super-pixel segmentation [FH04] to find a common homography for each superpixel, eliminating outliers via RANSAC [FB81]. The latter estimates several homographies from all matches in the images and assigns pixels to motion layers based on their brightness. Both sacrifice accuracy of the correspondences to a clear motion segmentation.

In our approach to integrate sparse and dense correspondences, we focus on highly accurate dense correspondence fields and incorporate features into optical

flow estimation. To achieve robustness, consistency and high accuracy we consider three neighboring images in the feature matching and in the optical flow estimation stage.

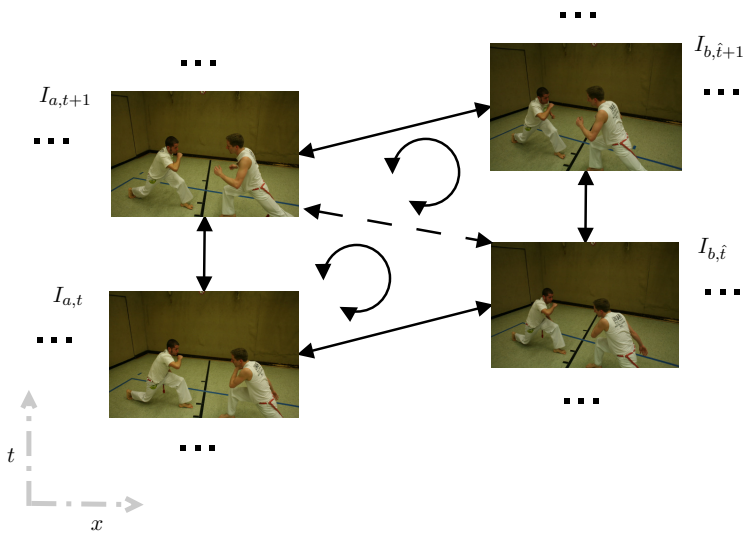


Figure 3.2: Four images of a multi-view video sequence: our algorithm accepts full-frame synchronized, uncalibrated images. We calculate dense correspondences which are consistent in the temporal direction t and the spatial direction x.

3.3 Three-Image Optical Flow

In our setup two or more video cameras provide uncalibrated and not necessarily perfectly synchronized image sequences. Usually, two-image optical flow is calculated independently between pairs of spatially or temporally neighboring images, [LLB+10, ZT03]. In our approach, we consider the flow between three neighboring images simultaneously, Fig. 3.2. Thus we can consider spatial and temporal neighbors at the same time and enforce consistency between them already during optical flow estimation. We refer to a triple of neighboring images as $I_1 : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$, $I_2 : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ and $I_3 : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ and to the forward flow between images I_i and I_j as $\mathbf{w}_{i,j} : \Omega \rightarrow \Omega$. We build our three-image based approach on the optical flow framework by Zach et al. [ZPB07] which yields

high-quality results and whose variants range among the top ten on the Middlebury evaluation site [BSL+07]. The basic idea of the approach is also used in the variational approach to alternate exposure imaging. In contrast to Sect. 2.6, we use it here together with the brightness constancy assumption as pointwise error term and update it between three spatially and temporally neighboring images simultaneously to obtain consistent optical flow on multi-view video sequences.

3.3.1 Duality Based Two-Image Optical Flow

The basic idea of the approach of Zach et al. is to split the pointwise data-term of optical flow estimation and the smoothness term using an auxiliary variable. The splitting and the resulting two equations, i.e., the pointwise minimization problem

$$\min_{\mathbf{v}\in\mathbb{R}^n} \lambda\ \psi(\rho(\mathbf{v})) + \frac{1}{2\theta}|\mathbf{u}-\mathbf{v}|_2^2, \tag{3.1}$$

for a fixed variable $\mathbf{u} \in \mathbb{R}^n$ and the image denoising problem

$$\min_{u_i\in\mathbb{R}} \int_\Omega \frac{1}{2\theta}(u_i - v_i)^2 + |\nabla u_i|\ d\mathbf{x}, \tag{3.2}$$

for $i \in \{1,\dots,n\}$ and a fixed function $\mathbf{v} : \Omega \to \mathbb{R}^n, \mathbf{v}(\mathbf{x}) = (v_1,\dots,v_n)^\top$, are detailed in Sect. 2.6.3, together with the dual solution scheme for the latter problem. Here, we look more closely at different data-terms $\rho(\mathbf{v})$ and replacements for the total variation regularization $|\nabla u_i|$. We discuss the motivation for these replacements and for the formulations employed in our algorithm we derive the iterative scheme necessary for the practical flow estimation.

Starting point of the discussion is the setup of Zach et al. that uses the absolute value of the brightness constancy assumption $|I_1(\mathbf{x}) - I_2(\mathbf{x}+\mathbf{w}_{1,2}(\mathbf{x}))| \approx 0$ as data-term and the total variation of the flow field as regularization, Sect. 2.6.1. As their framework turns out to be fast, efficient and flexible, admitting separate manipulations in data-term and regularization, it was subsequently adopted by several authors, who propose different regularizations and data-terms. Werlberger et al. maintain the brightness constancy as pointwise error and use the

thresholding scheme of Zach et al. to solve Eq. (3.1) but propose anisotropic regularization [WTP+09] that considers the image gradient when regularizing the flow field. As this regularization yielded more accurate results in preliminary experiments, we are going to use it in our framework and therefore have a closer look at this approach here. Instead of the total variation based image denoising problem, Eq. (3.2), they consider a denoising problem based on an anisotropic Huber-norm,

$$\min_{u_i \in \mathbb{R}} \int_\Omega \frac{1}{2\theta}(u_i - v_i)^2 + \psi_\epsilon((D^{\frac{1}{2}}\nabla u_i)^\top D^{\frac{1}{2}}\nabla u_i)\ d\mathbf{x} \tag{3.3}$$

where for the image gradient direction $\mathbf{g} = \frac{\nabla I_1}{\|\nabla I_1\|}$, $\mathbf{g}^\perp$ a unit vector perpendicular to $\mathbf{g}$ and constants $\alpha, \beta > 0$

$$D^{\frac{1}{2}} = e^{-\alpha\|\nabla I_1\|^\beta}\mathbf{g}\mathbf{g}^\top + \mathbf{g}^\perp(\mathbf{g}^\perp)^\top \tag{3.4}$$

is an image driven diffusion tensor and

$$\psi_\epsilon(s^2) = \begin{cases} \frac{s^2}{2\epsilon} & |s| \leq \epsilon \\ |s| - \frac{\epsilon}{2} & \text{else} \end{cases} \tag{3.5}$$

a robust penalizer that approximates the absolute value in a differentiable fashion. The anisotropic diffusion tensor attenuates the impact of the smoothing in the direction of the image gradient while maintaining the smoothing effect perpendicular to it. Thus smoothing over object and motion boundaries with high image gradients is prevented. Similarly to Eq. (3.2) the modified equivalent, Eq. (3.3), can be solved with a dual approach via

$$u_i = v_i + \theta\ \mathrm{div}(D^{\frac{1}{2}}\mathbf{p}) \tag{3.6}$$

where for $\tau = \frac{1}{4+\epsilon}$ the 2 dimensional vector $\mathbf{p}$ can be determined iteratively via

$$\mathbf{p}^{m+1} = \frac{\mathbf{p}^m + \tau(D^{\frac{1}{2}}\nabla u_i - \epsilon\mathbf{p}^m)}{\max\{1, |\mathbf{p}^m + \tau(D^{\frac{1}{2}}\nabla u_i - \epsilon\mathbf{p}^m|\}}\ , \tag{3.7}$$

starting with $\mathbf{p}^0 = \mathbf{0}$.

Making use of the dual approach to the image denoising problem, Eq. (3.2) or Eq. (3.3), respectively, Steinbrücker et al. [SPC09] and Linz et al. [LLM10] admit non-differentiable data-terms and perform a computational intensive full search in the image domain to solve the pointwise problem, Eq. (3.1). Although capable to detect large motion of small objects, their approach looses the real time capability of the thresholding scheme employed by Zach et al. and Werlberger et al.

A different modification of the data-term is proposed by Wedel et al. [WPB+09], who extend the thresholding scheme to include a further linear data-term, i.e., a fundamental matrix prior, requiring 3 more thresholding steps to include the additional constraint but achieving reasonable convergence speed. In Sect. 3.5.1 we include feature matching into optical flow estimation via local priors. However, even the extended the thresholding scheme cannot be used directly to accept locally defined flow as a prior. Therefore we replace the absolute value of the brightness constancy with a differentiable penalizer. For its simplicity we chose the squared value of the data-term. Thus, in our approach we replace the general data-term $\psi(\rho(\mathbf{v}))$ in Eq. (3.1) with $(I_1 - I_2(\mathbf{x}+\mathbf{w}_{1,2}))^2$. Substituting the general function $\mathbf{v} : \Omega \to \mathbb{R}^n$ with our forward flow field $\mathbf{w}_{1,2} : \Omega \to \mathbb{R}^2$ and the general function $\mathbf{u} : \Omega \to \mathbb{R}^n$ with the auxiliary flow field $\mathbf{u_{1,2}} : \Omega \to \mathbb{R}^2$ we can rewrite the minimization problem in Eq. (3.1) as

$$\min_{\mathbf{w}_{1,2}\in\mathbb{R}^2} \lambda \, (I_1 - I_2(\mathbf{x} + \mathbf{w}_{1,2}))^2 + \frac{1}{2\theta}|\mathbf{w}_{1,2} - \mathbf{u}_{1,2}|_2^2. \tag{3.8}$$

The insertion of the quadratic data-term renders the estimation of the flow field particularly simple: We iteratively determine estimates $\mathbf{w}_{1,2}^k$ for the flow field. Initializing with $\mathbf{w}_{1,2}^0 = \mathbf{0}$ we determine the next estimate from the current estimate via

$$\mathbf{w}_{1,2}^{k+1} = \mathbf{w}_{1,2}^k + d\mathbf{w}_{1,2} \ . \tag{3.9}$$

Thereby, the update $d\mathbf{w}_{1,2} = (dw_{1,2,1}, dw_{1,2,2})^\top$ is obtained by one step in a gradient descent framework. For this we apply a first degree Taylor expansion around

the current estimate $\mathbf{w}_{1,2}^k$ and obtain a quadratic function in $d\mathbf{w}_{1,2}$

$$E_q = (I_1 - I_2(\mathbf{x} + \mathbf{w}_{1,2}^k) + \nabla I_2 \; d\mathbf{w}_{1,2})^2 + \frac{1}{2\theta}|\mathbf{w}_{1,2}^k + d\mathbf{w}_{1,2} - \mathbf{u}_{1,2}|_2^2 \qquad (3.10)$$

from the minimization problem in Eq. (3.8). Setting $\frac{\partial E_q}{\partial d\mathbf{w}_{1,2,n}} = 0$ for $n \in \{1, 2\}$ we solve the resulting 2×2 linear system for the update $d\mathbf{w}_{1,2}$.

An experimental comparison for the optical flow algorithm with the absolute value of the brightness constancy and the squared value as data-term is given in Sect. 3.3.3.

Although the two-image optical flow algorithm performs very well on standard optical flow test-data, it does not yield results that are consistent on multi-view video sequences. In the next section we introduce additional constraints to enforce consistency.

3.3.2 Additional Consistency Constraints

In their approach Alvarez et al. restrict the solution space of optical flow estimation by enforcing symmetry between forward and backward flow [ADPS07]. If a point in the first image I_1 does not become occluded, following its flow to the second image I_2 and then returning with the backward flow from I_2 to I_1 should end at the starting point. Denoting the flow between I_2 and I_1 as $\mathbf{w}_{2,1}$ the symmetry constraint can be written as

$$\rho_s(\mathbf{x}) := \mathbf{w}_{1,2}(\mathbf{x}) + \mathbf{w}_{2,1}(\mathbf{x} + \mathbf{w}_{1,2}) \approx \mathbf{0}. \qquad (3.11)$$

If a point is occluded, the symmetry error for the flow fields at this point will be large, thus allowing for occlusion detection [ADPS07] as well as detection of spurious flows. Still, this approach considers only two images. Given three neighboring images, i.e. I_1, I_2 and I_3, we can consider a further constraint: for a point that is visible in all three images, a loop from I_1 over I_2 and I_3 going back to I_1 should end at the starting position. This constraint can be expressed as

$$\rho_l(\mathbf{x}) := \mathbf{w}_{1,2}(\mathbf{x}) + \mathbf{w}_{2,3}(\mathbf{x} + \mathbf{w}_{1,2}) + \mathbf{w}_{3,1}(\mathbf{x} + \mathbf{w}_{1,2} + \mathbf{w}_{2,3}) \approx \mathbf{0}. \qquad (3.12)$$

Occluded points are expected to possess also a high loop error as they cannot satisfy the constraint.

For regions where the optical flow is estimated to be a good approximation to the actual 2D motion field, both the symmetry and the loop error are expected to be small. Optical flow that results from spurious influences such as the lack of correspondences for occluded or noisy pixels will be detected via the loop error.

Eqs. (3.11) and (3.12) provide us with two error measures that can be applied in different ways. First, we can evaluate ρ_s and ρ_l as a confidence measure after the optical flow estimation on a sequence is terminated. Similar to the two and four image checks in Refs. [BW93, BJK08], the confidence values can be used to sparsify the optical flow, maintaining only those flow vectors that are judged reliable. We evaluate symmetry and loop error as a confidence measure in Sect. 3.3.4 to better understand where they are large, but in our approach we are actually interested in dense flow fields.

A second way to employ the symmetry and the loop constraint is as an additional data-term. This enforces the constraints already during the estimation of the flow. In contrast to the use of symmetry and loop error as post-estimation confidence measure, it returns dense flow fields. Yet, initially all flow fields are unknown. We therefore did not find it advantageous to require symmetry and loop-consistency as a hard constraint in the data-term.

Instead, we choose a third way and adapt the following strategy: For the flow field $\mathbf{w}_{i,j}^k$ we calculate the update based on the brightness constancy according to Eq. (3.10). Depending on how well the update satisfies the symmetry and the loop-consistency constraint Eqs. (3.11), (3.12) we set

$$\mathbf{w}_{i,j}^{k+1} = \mathbf{w}_{i,j}^k + c\, \mathbf{dw}_{i,j}, \tag{3.13}$$

where $c = \psi_1(\rho_s)\psi_2(\rho_l)$ with $\psi_i(\mathbf{z}) = \exp(\frac{-|\mathbf{z}|_2^2}{d_i})$ and parameters $d_1, d_2 > 0$. The function ψ_i thereby ensures that an update that satisfies the corresponding constraint perfectly, i.e., takes the value $\mathbf{0}$, is fully accepted with $\psi_i(\mathbf{0}) = 1$ while updates with large errors are rejected, i.e. $\lim_{\mathbf{z}\to\pm\infty} \psi_i(\mathbf{z}) = 0$. The parameters

$d_1, d_2 > 0$ steer the range of acceptable errors and are evaluated in more detail in Sect. 3.3.3.

The estimate $\mathbf{w}_{i,j}^{k+1}$ is used to calculate a regularized version $\mathbf{u}_{i,j}^{k+1}$ with the image driven dual approach from Ref. [WTP+09] given in Eqs. (3.6) and (3.7).

With the proposed iterative scheme, flow fields are only updated by the data-term where symmetry and the consistency on the loop are within an acceptable range. E.g. for occluded points where consistent updates are impossible and any brightness constancy-based update would lead to erroneous correspondence estimations, the updates are suppressed by the consistency check and, instead, neighboring valid flow is imposed via the smoothness constraint.

After updating $\mathbf{w}_{i,j}^{k+1}$ all other unknown flow fields between the images I_1, I_2 and I_3 are updated, before the next update $\mathbf{dw}_{i,j}$ is determined and checked for consistency. To speed up calculations and to prevent temporal aliasing, we implement our algorithm to work on an image pyramid of factor 0.5. As all flow fields are upscaled simultaneously, consistency checks can be performed on all levels as described above. We also keep the constants d_1 and d_2 fixed over all levels in spite of the changing resolution, as we want to allow the flow on coarse levels to exploit their neighborhood more freely than on fine levels.

If I_1, I_2, I_3 are all acquired by one camera, constraint (3.12) is similar to the temporal smoothness constraint for optical flow [BA91]. It ensures consistency within a sequence captured by one camera, but it does not relate images acquired by two cameras. A more suitable assignment is, for instance, to choose $I_1 = I_{a,t}$ and $I_2 = I_{a,t+1}$ to be consecutive images acquired by the camera C_a and $I_3 = I_{b,\hat{t}}$ an image acquired by camera C_b at approximately the same instant as I_1, Fig. 3.2. If in a multi-view video sequence more than three images are given, we calculate the optical flow for each triple independently. Thus we achieve to estimate flow robustly and consistently on three images, but avoid to propagate errors through the sequence.

3.3.3 Evaluation of Three-Image Consistent Optical Flow

For the evaluation of our algorithm we calculate the optical flow on datasets for which the ground-truth motion is known. This allows to compare the estimated flow fields to the ground-truth motion and to determine the average angular error (AAE) and the average endpoint error (AEE) as accuracy measures. Standard test scenes for optical flow [BFB94, BSL+07] are only of limited suitability for our evaluation as they only provide a sequence of images acquired with one camera. In this case our algorithm degenerates to enforcing temporal coherence in an unnecessary complicated way. To show that our algorithm is applicable also to these datasets, we include two real-image sequences of the Middlebury optical flow dataset [BSL+07], *rubber whale* and *hydrangea* in our tests, Fig. 3.4. All images in these two test scenes are 388×584 pixels, the *rubber whale* scene contains displacements of up to 5 pixels per frame and the *hydrangea* scene contains motion of up to 12 pixels per frame. We also use the Middlebury stereo datasets [SS02] *art*, *books*, *dolls*, *moebius* with motion up to 113 pixels per frame on 555×695 pixel images and the datasets *laundry* and *reindeer* with motion up to 117 pixels per frame on 555×671 pixel images, Figs. 3.3 and 3.4. These sequences simulate a stereo camera moving around a rigid scene, but we do not exploit the stereo information in our algorithm. To evaluate the performance of our algorithm also on the non-rectified, non-synchronized images for which it is intended, we created two additional test-sequences, *waving* and *stonemill*, Fig. 3.4. The sequences contain images with 600×800 pixels and 270×480 pixels, respectively, and ground-truth motion of up to 40 pixels. The datasets are available on our web-page `http://www.cg.cs.tu-bs.de/projects/datasets` for the evaluation of space-time consistent correspondence algorithms. The ground-truth motion in Figs.3.3(c) and 3.4(c) are color-coded with the color-map from Ch.2, Fig. 2.12(a), p. 39. Additionally, pixels for which ground-truth data is not available are marked black. To obtain better exploitation of the color-scale in the stereo sequences, in which all pixels have negative disparity, we subtract the minimal disparity from

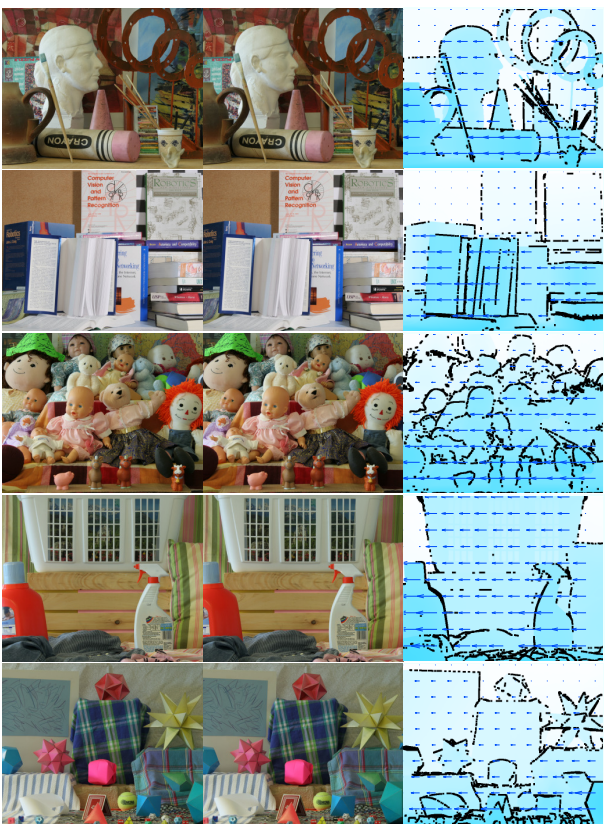
all flow fields for the visualization so that the smallest ground-truth disparity is displayed as white.

Throughout all experiments we update the flows $\mathbf{w}_{i,j}^k$ 10 times, i.e. $k = 0 \dots 9$ working on an image pyramid with 6 levels and updating the dual variables $\mathbf{p}$ in Eq. (3.7) with two iterations according to Ref. [WTP+09]. For normalized intensity values we determine suitable parameters $\lambda \in [10, 1000]$, $\theta \in (0, 2]$ and $d_1, d_2 \in [0.5, 100]$ for each dataset experimentally. Using unoptimized MATLAB prototype code the estimation of all six flow fields between three neighboring images of 270×480 pixels with the basic gradient descend approach last 149 s on a 2.66 GHz processor. Inclusion of the constancy checks increases computation time to 189 s, i.e., to 1.27 times the basic computation time.

The Influence of the Consistency Checks

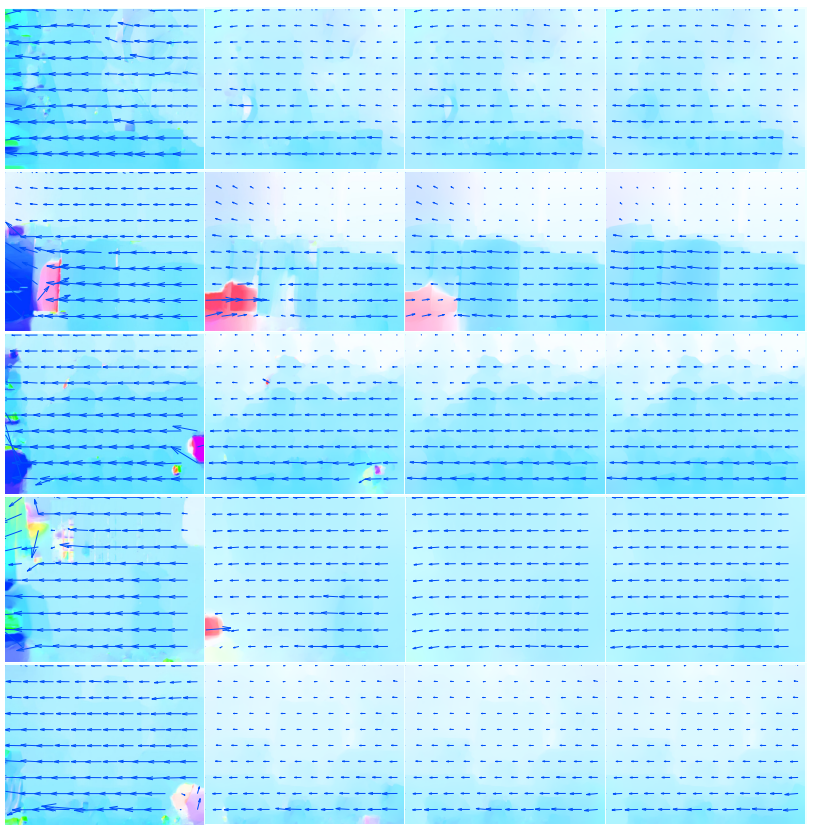
In the derivation of our symmetric and loop-consistent flow we changed several formulations starting from the approach of Werlberger et al. [WTP+09]. We evaluate all modifications experimentally to demonstrate their effect on the estimated flow fields. For comparison, we first estimate flow fields using the absolute value of the data-term and the thresholding scheme as they are originally proposed by Werlberger et al. Secondly, we calculate flow fields using the squared value of the data-term and the gradient descend scheme proposed in Sect. 3.3.1 which we subsequently call Huber-L^2 approach. This is the basic algorithm into which we include our consistency checks. We then calculate the flow fields restricting the update to satisfy the symmetry constraint, Eq. (3.11). In the final step of our accuracy evaluation, we enforce both symmetry and loop-consistency, Eqs. (3.11) and (3.12).

As Tab. 3.1 shows, the approach of Werlberger with the absolute value data-term obtains optical flow estimates that have a smaller average angular error and average endpoint error in some of the scenes, while on the Middlebury stereo scenes the squared data-term performs better. Comparing the flow fields visually,



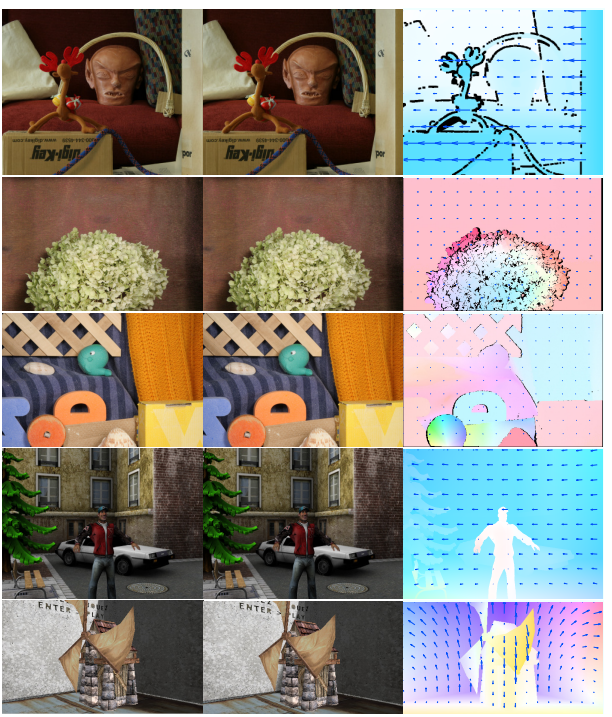
(a) (b) (c)

Figure 3.3: The datasets *art*, *books*, *dolls*, *laundry* and *moebius* used in our flow evaluation: (a),(b) the first two input images I_1 and I_2 and (c) the ground-truth flow field between them (color-coded with the color-map in Fig. 2.12(a) and black for unknown pixels).



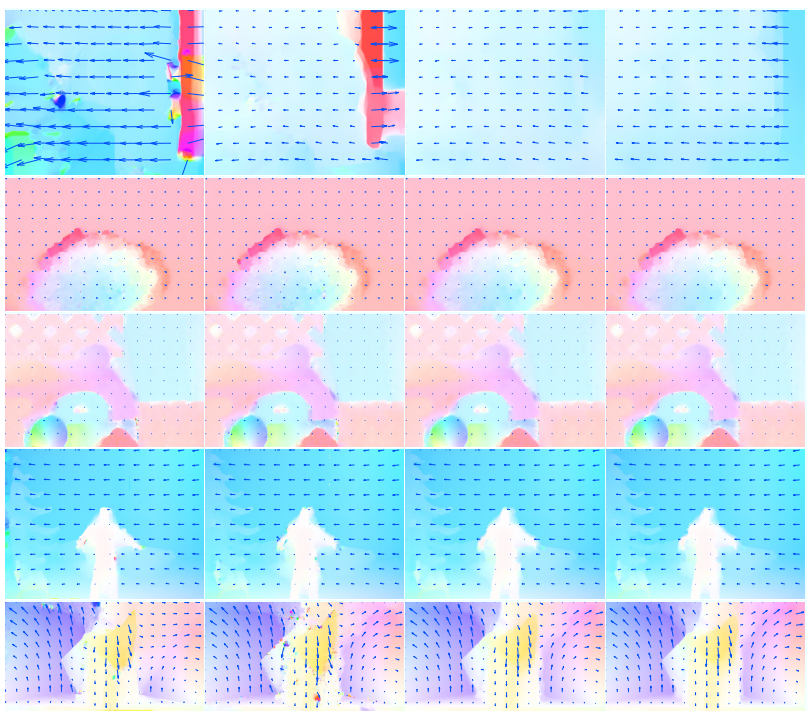
(d) (e) (f) (g)

Figure 3.3: (cont.) The estimated flow fields for the datasets *art*, *books*, *dolls*, *laundry* and *moebius* (color-coded, see Fig. 2.12(a)): (d) Using the thresholding scheme of the Huber-L^1 approach [WTP+09] is not convincing for these datasets. (e) The result of the Huber-L^2 algorithm can be improved by imposing (f) symmetry and (g) consistency on a loop during flow field estimation.



(a) **(b)** **(c)**

Figure 3.4: The datasets *reindeer*, *hydrangea*, *rubber whale*, *waving* and *stonemill* used in our flow evaluation: (a), (b) the first two input images I_1 and I_2 and (c) the ground-truth flow field between them (color-coded with the color-map in Fig. 2.12(a) and black for unknown pixels).



(d) (e) (f) (g)

Figure 3.4: (cont.) The estimated flow fields for the datasets *reindeer*, *hydrangea*, *rubber whale*, *waving* and *stonemill* (color-coded, see Fig. 2.12(a)): (d) Using the thresholding scheme of the Huber-L^1 approach [WTP+09] shows acceptable results but is inflexible with respect to manipulations of the data-term. (e) The result of the Huber-L^2 algorithm can be improved imposing (f) symmetry and (g) consistency on a loop during flow field estimation.

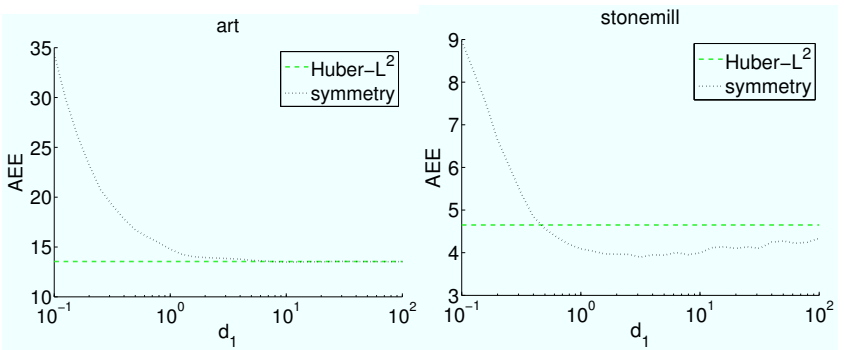
	[WTP+09]		Huber-L^2		Symmetry		Loop	
	AAE	AEE	AAE	AEE	AAE	AEE	AAE	AEE
rubber whale	5.52°	0.17	6.01°	0.19	5.85°	0.18	5.64°	0.17
hydrangea	2.66°	0.23	2.78°	0.25	2.67°	0.24	2.58°	0.23
art	2.39°	14.54	1.07°	13.54	1.04°	13.52	0.88°	11.22
books	11.39°	22.88	10.63°	15.67	3.08°	10.45	1.34°	4.31
dolls	4.99°	17.03	0.76°	3.72	0.35°	2.7	0.35°	2.62
laundry	6.39°	15.95	3.23°	14.21	1.49°	11.84	0.87°	10.27
moebius	5.02°	8.49	0.69°	4.58	0.61°	4.47	0.55°	4.42
reindeer	22.38°	40.34	17.99°	25.19	1.63°	18.8	0.93°	12.88
waving	2.63°	1.31	2.79°	0.99	2.67°	0.96	2.48°	0.9
stonemill	17.39°	5.41	17.59°	4.65	12.72°	3.97	10.43°	3.53

Table 3.1: Comparison to ground-truth motion: In some cases the Huber-L^1 motion field estimation [WTP+09] has a smaller average angular (AAE) and smaller average endpoint error (AEE), in some cases the Huber-L^2 estimation. As the latter is easier to manipulate, we implement our consistency check into the Huber-L^2 framework. The symmetry check decreases the AAE and the AEE of the basic Huber-L^2 approach. Enforcing consistency on a loop decreases the error in both measures further.

Figs. 3.3 and 3.4, we remark that the approach with the absolute value data-term tends to assign outlier motion to some regions where deviations in the data-term were not sufficiently strong to guide the smoothed flow to the correct regions. Note though that all test scenes contain no or only little noise. The absolute value of the data-term was introduce to optical flow estimation to deal more robustly with noise than the squared data-term. We mainly adopted the differentiable squared data-term to allow for easier modification in Sect. 3.5.

Comparing the resulting errors of the basic approach with squared data-term to the approach that additionally considers symmetry, Tab. 3.1, shows that both the average angular error and the average endpoint error are reduced in all scenes. The flow fields with the symmetry constraint in Figs. 3.3(f) and 3.4(f) are less noisy and have fewer outliers than the flow fields obtained without the constraint, Figs. 3.3(e) and 3.4(e), although the common parameters λ and θ were maintained fixed. Including the loop-consistency check removes the remaining outliers in the *books* sequence and yields generally visually promising flow fields, Figs. 3.3(g) and 3.4(g). Evaluation of the average angular error and the average endpoint error relative to the ground-truth motion shows, Tab. 3.1, that both can be decreased, i.e., the accuracy of the flow fields is improved. We observe that even the error of the basic squared data-term on the monocular sequences *hydrangea* and *rubber whale* can be slightly improved by enforcing consistency. Most important are the improvements in the stereo sequences, where the consistent optical flow is able to detect the large motion reliably.

Apart from the parameters λ and θ inherited from the approach of Werlberger, our modifications introduce two new parameters, d_1 and d_2, that steer the range of the accepted symmetry and loop errors in the update. If for a fixed symmetry error ρ_s the parameter d_1 is chosen very small, the corresponding weight ψ_1 is very small and the update in Eq. (3.13) is suppressed. In contrast, if d_1 is large, ψ_1 is close to 1. The dependency of the average endpoint error on the actual value of d_1 is shown in Fig. 3.5. We observe that once d_1 is sufficiently large, the influence of the value d_1, scaled logarithmically in the plots, is small. If the value



(a) (b)

Figure 3.5: The choice of the parameter d_1 which steers the influence of the symmetry error in the symmetric flow influences the average endpoint error (AEE) of the estimated flow fields. For the shown datasets (a) *art* and (b) *stonemill* as representative test-cases, the parameter d_1 must be chosen sufficiently large to allow for updates. Once the critical value for d_1 is passed, the AEE of the symmetric flow (black, dotted) drops below that of the basic Huber-L^2 approach (green, dashed).

d_1 is chosen too small, all updates of the flow field are suppressed. If a certain deviation in the symmetry is allowed, only gross symmetry errors are suppressed and the flow converges to better results than without the symmetry check.

Evaluating the influence of the parameter d_2 that steers the range of accepted loop errors in the update, Fig. 3.6, we observe, as for d_1, that the constant should not be chosen too small to suppress all updates, but that values larger than 1 produce good results.

Evaluation of the Consistency on a Stereo Sequence

The general idea of our approach is to enforce consistency on multi-view video sequences. Therefore, we also evaluate the differences in the end position obtained by following a pixel's flow first in time within the video stream of camera C_a and then to the video stream of camera C_b and by following the flow first to the other

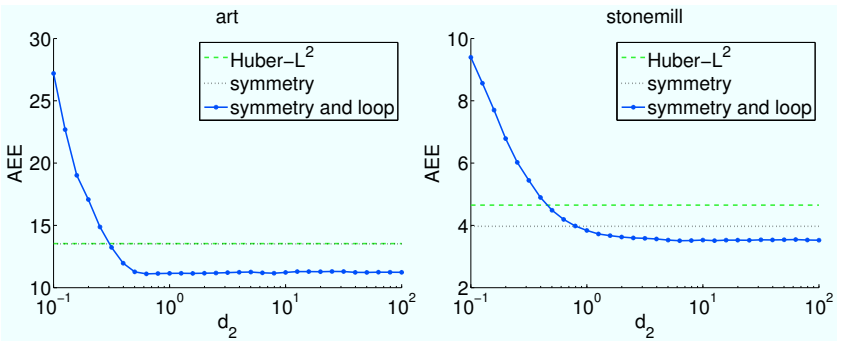


Figure 3.6: The choice of the parameter d_2 which steers the influence of the loop error in our loop-consistent flow influences the average endpoint error (AEE) of the estimated flow fields. For the shown datasets (a) *art* and (b) *stonemill* as representative test-cases, the parameter d_2 must be chosen sufficiently large to allow for updates. Once the critical value for d_2 is passed, the AEE of the loop-consistent flow (blue, solid with dots) drops below that of the basic Huber-L^2 approach (green, dashed) and the approach using only symmetry (black, dotted).

video stream and then in time, Fig. 3.1. If the basic two-image approach is applied or only symmetry is imposed, the endpoint positions depend on the direction of processing, Tab. 3.2. However, if consistency in a loop of three images is imposed, the difference in position is decreased. Again we observe that the improvement of the three-image flow is more pronounced in sequences with large motion like *dolls* and *reindeer* and less visible in the monocular scenes with the small motion.

The advantage of this consistency check is that it can also be evaluated on real scenes. In addition to the scenes with known ground-truth, we therefore consider the scenes *Heidelberg* and *skydive*, both 720×1280 pixel stereo sequences [Wim10]. Furthermore we evaluate our algorithm on the multi-video recordings *market*, 421×452 pixel, *capoeira*, 817×578 pixel and *skateboard*, 270×480 pixel, which are recorded using unsynchronized, uncalibrated cameras, while in the scene *jump*, 270×480 pixel, cameras are additionally hand-held. These multi-view images

	Huber-L^2		Symmetry		Loop	
	APD	VPD	APD	VPD	APD	VPD
rubber whale	7.24	6.54	5.53	2.02	5.41	1.92
hydrangea	15.32	7.35	13.4	3.84	12.9	3.74
art	1.78	1.89	1.72	1.75	1.24	1.31
books	13.09	28.33	9.57	20.41	2.79	5.01
dolls	11.75	25.69	0.92	1.71	0.67	1.28
laundry	5.57	12.13	2.33	2.72	1.25	1.81
moebius	1.51	4.34	1.22	2.01	0.70	1.17
reindeer	10.75	21.38	10.91	10.97	3.98	3.4
waving	0.29	2.12	0.21	1.29	0.19	1.23
stonemill	2.53	8.74	0.88	1.85	0.69	1.52

Table 3.2: Comparison of different orderings to process spatially and temporally neighboring images: Imposing symmetry and consistency in a loop of three images yields smaller average position difference (APD) and a smaller variance in the position difference (VPD) between considering first spatial and then temporal neighbors and considering first temporal and then spatial neighbors than the basic Huber-L^2 approach or imposing symmetry only.

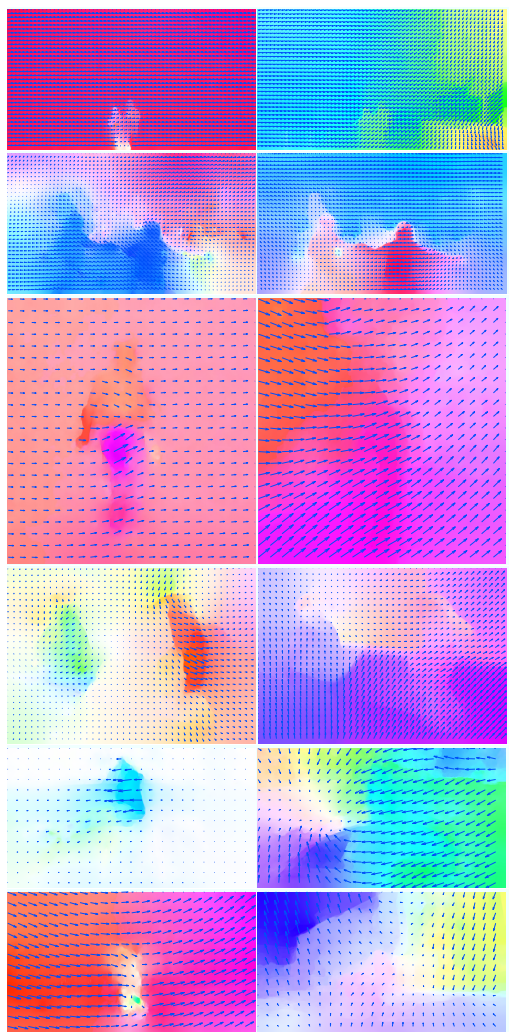
are recorded with cameras using automatic color-balancing and gain, so that as additional challenge, they are only poorly brightness adjusted. For the estimation of the flow fields on these scenes, we fix $\lambda = 200$, $\theta = 1.5$, $d_1 = 1$ and $d_2 = 1$ as they are reasonable parameters in all test scenes with known ground-truth and return comparatively good flow fields also for the real scenes, Fig. 3.7. As zero flow fields between all images would provide excellent consistency between the two concatenation directions, we first evaluate the flow fields visually to ensure that flow fields are estimated realistically. We observe that also on these scenes, the loop-consistency is able to correct the outlier flow regions assigned by the two-image approach such as the sky in the *jump* scene (last row) or around the left person in the *capoeira* scene while maintaining details such as the hands of the person in the *market* scene or the trees in the *Heidelberg* scene. Note that the two-image flow in the center of the *market* scene in Fig. 3.7(b) are spurious flows due to occlusion and disocclusion and do not correspond to the motion of the ball. Although not able to detect the large motion of the small ball correctly, the loop-consistent flow is able to remove the spurious flow at the occluded points. In addition to these improvements, Tab. 3.3 shows that also the consistency for proceeding first in time and then in space and for proceeding first in space and then in time is improved by our proposed algorithm.

3.3.4 Symmetry and Loop Error as Confidence Measures

We also evaluate the violation of the symmetry and the loop constraint as confidence measures for optical flow. On the one hand this helps us to understand which updates are suppressed during loop-consistent flow estimation. On the other hand this allows other authors to apply the constraint also with optical flow algorithms that do not allow to include the check directly into the flow estimation. If a certain parameter setting returns optical flow estimations that violate the symmetry and loop constraint on large regions, the algorithm can be

(a) (b) (c)

Figure 3.7: For the recordings (a) *Heidelberg*, *skydive*, *market*, *capoeira*, *skateboard* and *jump* we estimate flow fields between temporally and spatially neighboring images with (b),(c) the basic Huber-L^2 algorithm and (d),(e) our consistent optical flow (color-coded with the color-map in Fig. 2.12(a)).



(d) (e)

Figure 3.7: (cont.) (d),(e) Results of our consistent optical flow.

	Huber-L^2		Symmetry		Loop	
	APD	VPD	APD	VPD	APD	VPD
Heidelberg	0.22	0.59	0.19	0.32	0.18	0.28
skydive	0.49	0.89	0.42	0.58	0.37	0.49
market	6.79	9.37	6.09	4.43	3.37	2.06
capoeira	13.61	14.92	9.72	9.31	5.17	3.45
skateboard	1.76	1.84	1.62	1.42	1.17	0.95
jump	4.64	6.61	4.34	1.78	2.96	0.99

Table 3.3: Consistency between considering first spatial and temporal flow and then temporal and spatial flow can also be evaluated for scenes with unknown ground-truth motion. Imposing symmetry and consistency in a loop of three images yields smaller average position difference (APD) and a smaller variance in the position difference (VPD) between the two processing orders than the basic Huber-L^2 approach or imposing symmetry only.

started again with a different set of parameters or the flow can be sparsified and inpainted in these regions.

In our evaluation we analyze two cases. In the first case, we consider flow fields that were estimated independently with the basic squared data-term algorithm, i.e., where the error measures were not enforced during optimization. In the second case, we consider flow fields that were updated with our proposed consistency check. For both cases, we compare the confidence measure obtained by the symmetry and the loop constraint to other confidence measures from literature. An often proposed confidence measure is related to the strength of the gradient ∇I_1, as the data-term for optical flow is not well defined at points with zero gradient [BFB94, NDB09]. So points with small gradient are assigned a very low confidence. In addition to this purely image based confidence measure, we compare our symmetry and loop based confidence to a confidence measure based on the optimization residual [BW06].

In the experiment we successively sparsify the estimated motion field based on the confidence measure under consideration [BW06]. For the remaining pixels

we evaluate the average endpoint error. The optimal confidence measure for this procedure is the pixelwise endpoint error. We also sparsify flow fields on the basis of the pixelwise endpoint error to obtain the curve for the optimal sparsifier.

Comparing the curves for the flow fields estimated with different algorithms, shown in Fig.3.8 in column (a) and (b) respectively, we observe that the three-image optical flow has a lower starting point but otherwise the method with which we estimate the flow has little influence on the relative course of the curves for the different confidence measures.

Comparing the different confidence measures, Fig. 3.8, we observe that the confidence measure based on the image gradient performs quite poorly in all scenes. Fig. 3.8 also shows that in the crucial, initial stage of the first 20% of sparsification either the loop or the symmetry based confidence measure is closer to the optimal curve than the optimization residual. For some of the datasets, e.g. *laundry* and *stonemill*, symmetry and loop based confidence is a fairly close approximation to the optimal curve in this range.

Images of the strength of the confidence measures are shown in Fig. 3.9 for the basic Huber-L^2 approach and in Fig. 3.10 for our loop-consistent approach. We normalized the images for each confidence measure separately, dividing by the largest error value obtained with both algorithms so that the two images for the same confidence measure are comparable. Considering these two figures permits to make several observations. As already found in the determination of the mean angular and the mean endpoint error, Tab. 3.1, the loop-consistent flow eliminates some regions of gross endpoint error, such as the lower left corner in the *books* scene (third row) or at the right side of the *dolls* scene (fourth row). Column 3.9(d) and 3.9(c) confirm that these regions are also assigned a high symmetry and loop error which can be considerably reduced by the consistent flow estimation, Fig. 3.10(d) and (c). Generally, symmetry and loop based confidence measures can detect errors due to occlusion effects quite successfully as can be best seen around the arms of the person in the *waving* scene and the wings of the mill in the *stonemill* scene. Additionally we observe that large endpoint errors at the

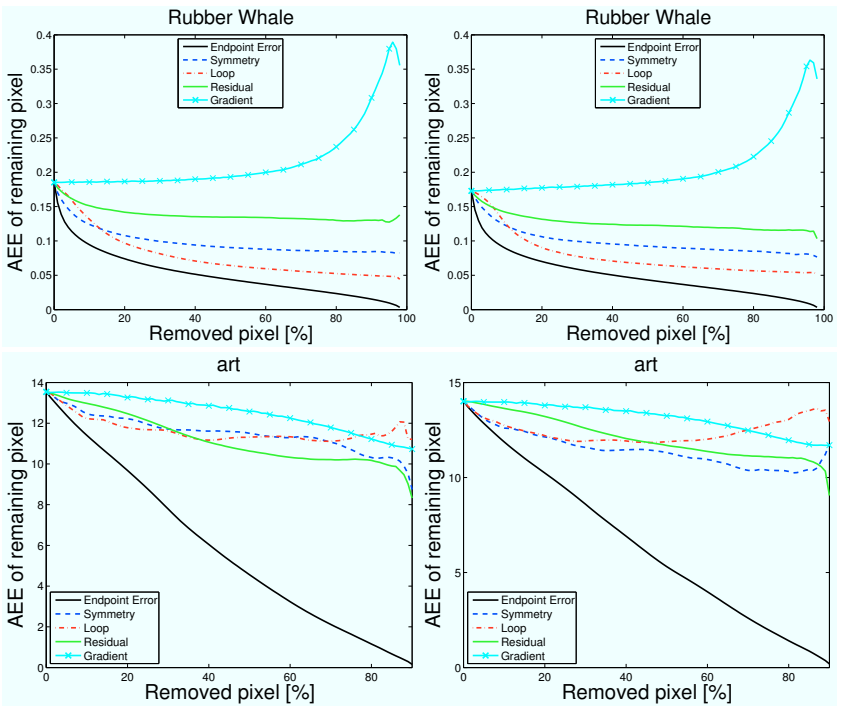
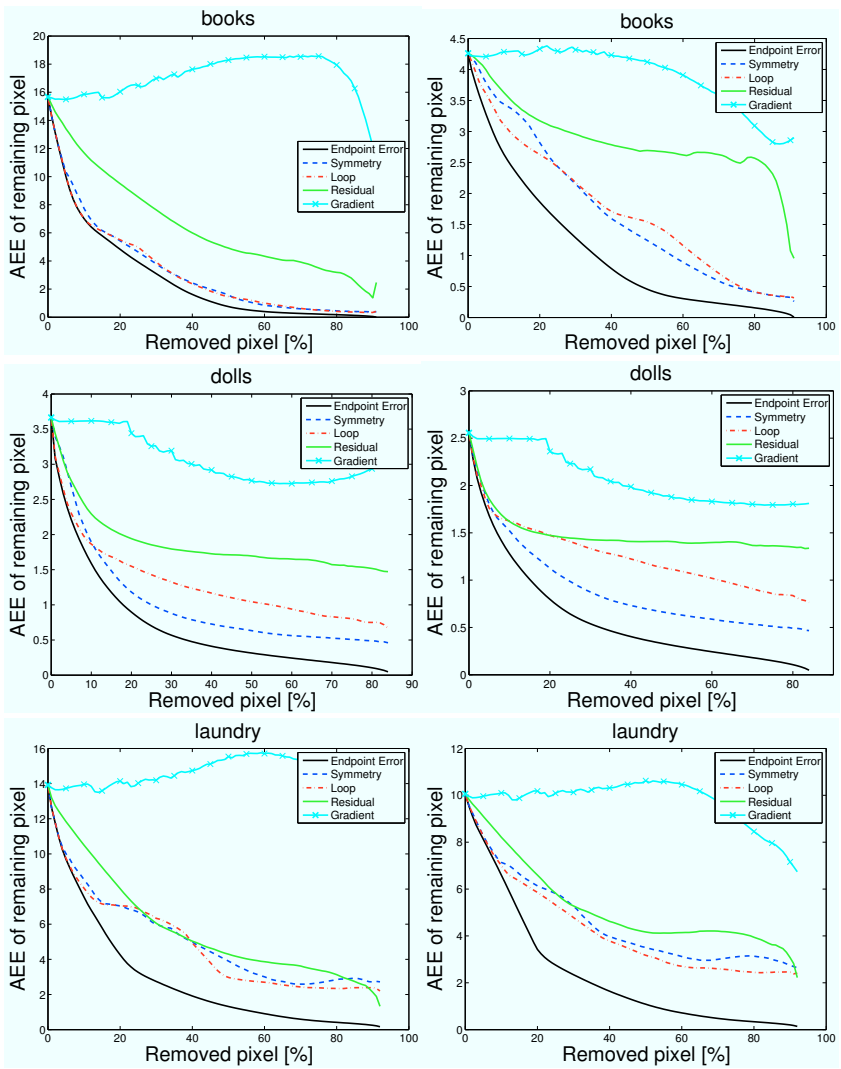


Figure 3.8: Flow fields estimated with (a) the basic Huber-L^2 approach and (b) our symmetry and loop-consistent flow are sparsified successively with different confidence measures. The average endpoint error (AEE) of the remaining pixels is evaluated. In most cases, the curves obtained with the symmetry (blue, dashed) and the loop (red, dash-dot) measures are much closer to the optimal confidences measures (black, solid) than the curves obtained with the confidence measures based on the optimization residual (green, solid) while the image gradient has little correlation with the accuracy of the estimated flow (cyan, solid with crosses).



(a) (b)

Figure 3.8: (cont.) Sparsification with different confidence measures for the datasets *books*, *dolls*, *laundry*.

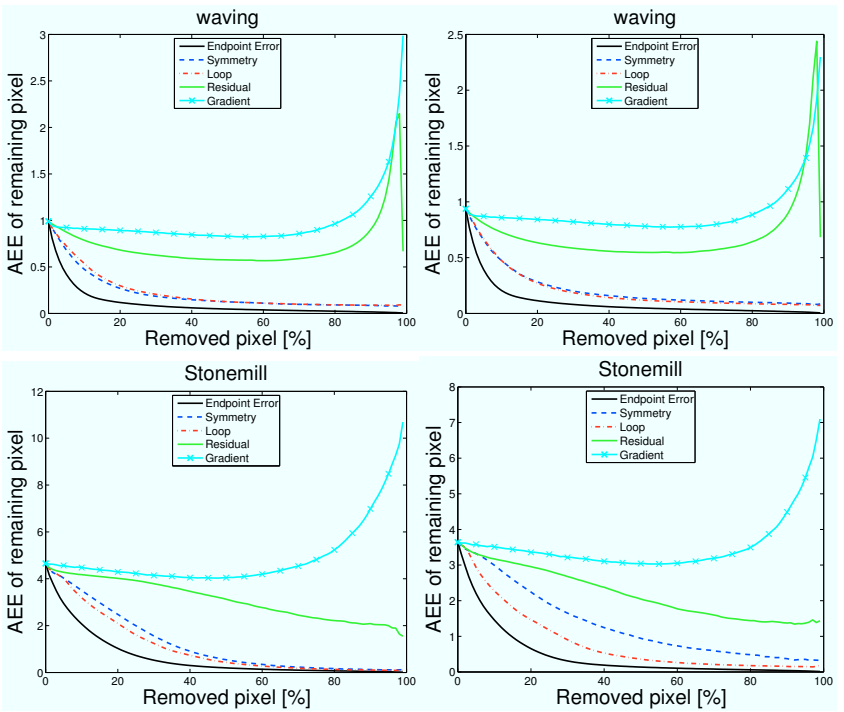


Figure 3.8: (cont.) Sparsification with different confidence measures for the datasets *waving* and *stonemill*.

occlusion borders of the *waving* and the *stonemill* scene, Fig. 3.9(a), are reduced for the loop-consistent optical flow, Fig. 3.10(a).

3.3.5 Discussion

Considering symmetry and loop-consistency of optical flow on three images provides us with additional constraints for optical flow estimation from multi-view video sequences. Employing these constraints in a consistency check for each iterative update during the flow estimation allows us to estimate more accurate flow

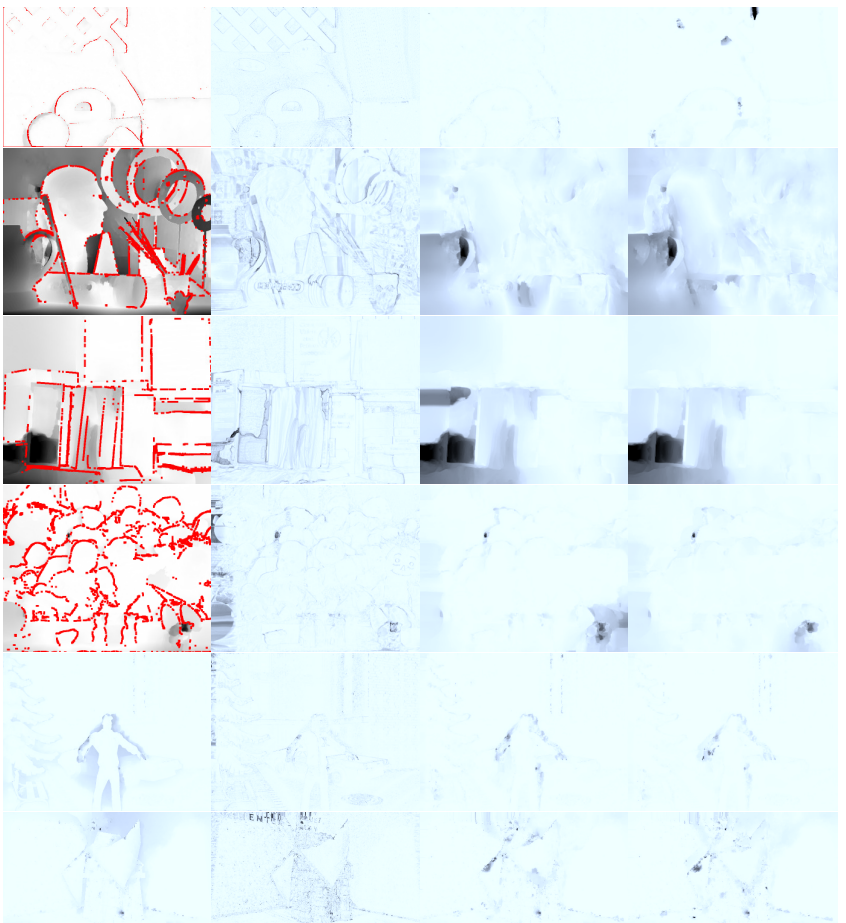
fields. Evaluation of the symmetry and loop constraint as confidence measure shows that the consistency checks detect points with high endpoint error quite reliably. The proposed framework suppresses flow updates that do not pass the consistency check in a continuous fashion and relies on the fill-in effect of the smoothness constraint to propagate reliable flow to these regions. As occluded points cannot pass the consistency checks, the algorithm also propagates reasonable flow estimates to occluded regions respecting objects boundaries due to the anisotropy of the regularization constraint.

In spite of a strong dependency on the fill-in effect, the algorithm does not require an increased number of iterations to produce results with higher accuracy. We run all estimations with as little as 10 iterations per level of the image pyramid.

The overall computational load of the proposed algorithm is slightly increased in comparison to estimating all flow fields between the three images independently.

For real scenes we also apply our optical flow between images that are acquired with different cameras and are therefore not perfectly color- and gain- adjusted. In spite of this additional challenge, our three-image consistent algorithm shows the same improving behavior in all flow fields between three spatio-temporal neighboring images.

In contrast to the alternate exposure approach, the three-image consistent flow relies on the successive low-pass filtering of the image pyramid to cope with large motion and thus cannot detect motion of isolated small objects. We will look into ways to solve this problem in the next sections.



(a) (b) (c) (d)

Figure 3.9: For the basic Huber-L^2 optical flow algorithm we compare the normalized strength of different confidence measures on the estimated flow fields with black indicating large deviations of the flow and white coincidence, red points have unknown ground-truth. (a) Comparing to the ground-truth motion, the endpoint error provides an optimal confidence measure. While (b) the optimization residual based confidence measure is sensible to single pixel outliers, (c) symmetry and (d) loop-based confidence measures detect regions of inaccurate flow more reliably.

(a) (b) (c) (d)

Figure 3.10: For our loop-consistent optical flow algorithm we compare the normalized strength of different confidence measures on the estimated flow fields. The same confidence measures as in Fig. 3.9 are evaluated.

3.4 Feature Matching on Three Images

As already discussed in Sect. 3.2, sparse feature matching is used as initialization for many algorithms in computer vision and computer graphics [SLW+10, BWSS09, BBM09]. This low-level processing step is able to establish coarse correspondences very quickly. Determination of robust feature points and corresponding feature point descriptions has been an intensely investigated area of research for decades [MTS+05, MS05]. In spite of great advances, wrongly matched correspondences are still commonly encountered. If additional information on the images is provided, e.g. by calibration, synchronization or assumption of constant rigid motion, this information can be used to eliminate wrongly matched correspondences [HZ03]. Unfortunately, in practical applications, additional information is not always available as, for instance, multiple cameras are hard to synchronize in an outdoor environment [MSMP08] and usually images of independently moving objects are recorded. The goal of this section is to develop a versatile, robust feature point matching method that is generally applicable, e.g., also in the unconstrained multi-view video setup. Similarly to the dense correspondence estimation in the previous section, we exploit the redundancy in the data of multi-view video sequences with a common field of view. We use the information from the additional images to establish more reliable correspondences to ensure high-quality matches. Feature points are matched by considering closed loops of images. We introduce three-image consistent matching and evaluate it by means of the percentage of wrong matches.

3.4.1 Three-Image Feature Matching

As in Sect. 3.3.2 let $I_1 : \Omega \to \mathbb{R}$, $I_2 : \Omega \to \mathbb{R}$ and $I_3 : \Omega \to \mathbb{R}$ be three images of a multi-view video sequence that have some common field of view on a dynamic scene. In contrast to previous robust matching methods, we do not require epipolar geometry between images to be applicable, nor do we assume a temporal ordering, i.e. the three images can be acquired by one, two or three

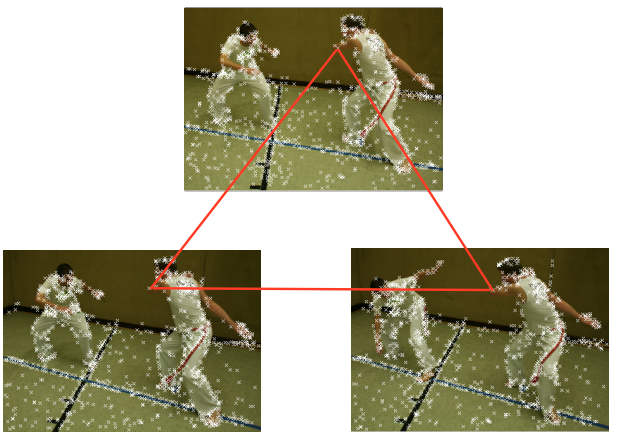


Figure 3.11: Three images of a multi-view video sequence with detected SIFT features: our algorithm accepts three images with some common field of view acquired by one or several unsynchronized and uncalibrated cameras. By matching features on three images simultaneously, false matches are eliminated and correspondences between images can be established more robustly.

unsynchronized cameras, Fig. 3.11. For each image I_i, $i \in \{1, 2, 3\}$ a feature detector determines features $f_{i,l}, l \in \{1, \ldots, N_i\}$ with corresponding descriptors $s_{i,l}$. In the descriptor space a distance function is required for feature matching which we denote as $d(s_{i,l}, s_{j,m})$. In our experiments, Sect. 3.4.2, we evaluate the algorithm for several detector/ descriptor variants, so we keep the description general in this section.

Usually, after detection the features are matched between two images at a time. Authors of different descriptors propose slightly different matching methods. To keep the results comparable, we follow the approach of [MS05] and use nearest neighbor matching (NN) for all two-matching steps in our evaluation. A more elaborate two-matching strategy (NNDR) compares the distance of the

nearest neighbor to the distance of the second nearest neighbor and only accepts a match if their ratio is below a threshold [Low04]. We additionally include this matching strategy into our evaluation.

If more than two images are considered, inconsistencies in the matches such as $(f_{1,l}, f_{2,m})$, $(f_{1,l}, f_{3,n})$ and $(f_{2,m}, f_{3,p})$, $p \neq n$ become obvious. In multi-view video, corresponding feature points are supposed to belong to one single scene point, so inconsistent matches indicate false matches. A straightforward approach to reduce the number of false matches is to filter out any match that is not consistent on a closed loop of three images. To eliminate inconsistent matches already during the assignment we formulate the matching problem in a different way.

In our approach we look for *triples* $(f_{1,l}, f_{2,m}, f_{3,n})$ such that each $f_{i,j}$ is present in at most one triple. To each of the triples we assign a cost $\tilde{d}$ that is the sum of the distances of all three descriptors

$$\tilde{d}(s_{1,l}, s_{2,m}, s_{3,n}) = d(s_{1,l}, s_{2,m}) + d(s_{2,m}, s_{3,n}) + d(s_{3,n}, s_{1,l}) \ , \tag{3.14}$$

i.e. the distance between each pair of features is considered in the cost function, which therefore is independent of the ordering of the images. In contrast to previous approaches this formulation requires the matches in all images to be similar and thus closes the loop between the images, providing a feedback to the matching and avoiding the drift commonly encountered in considering ordered sets of images. If all features were present in all three images, i.e., if a complete matching for sets of equal, finite cardinality was required, this is an instance of the classical three-matching problem with decomposable cost-function, an NP hard problem which can be solved approximately with the following algorithm [CS92]:

i. Match the features in I_1 and I_2, e.g., using the Hungarian algorithm, (see [PS98]).

ii. Merge the sets of features on the basis of the matching in (i.) such that the new cost function between features in I_1 and I_3 is $\hat{d}(s_{1,l}, s_{3,n}) = \tilde{d}(s_{1,l}, s_{2,m}, s_{3,n})$.

iii. Match the features in I_1 and I_3 with the new distance function.

iv. Sum up all distances present in the matching.

v. Interchange the role of I_1, I_2, I_3 and restart at (i.).

vi. Of the three matchings thus obtained, return the one with the smallest sum of distances.

Note that step (ii.) requires the third feature in the triple to be close both to the feature in I_1 and the feature in I_2. Enforcing this condition provides the means to transport the information of the two images already matched to the matching to the third image.

The three-match returned by this algorithm can be proved to lie within a certain distance to the actual best solution and in practice it often turns out to be the best solution [BCS94].

However, working with real images, we have to deal with occluded and non-detected features as well as with non-distinctive descriptors, i.e., we neither have feature sets of equal cardinality nor do we insist on matching all features. We therefore adjust the above algorithm. In step (i.) we use NN matching or optionally NNDR matching. Additionally, we match feature points only if they are mutual nearest neighbors, i.e., by imposing a symmetry constraint. Thus the processing is independent from the ordering of the images. For step (ii.) we remove all features from both images that are not matched in the previous step. We are only interested in feature points that can be matched consistently in three images. As the number of feature points differs in every image and we do not require all feature points to be matched, the sum of all matchings is no longer a reliable quality measure and step (iv.) is skipped. Correspondingly, for step (vi.)

we do not return the match with the smallest overall cost, as this is dependent on the number of feature points actually matched. Instead we merge the three matches and only return those triples that are found in all three matching directions. Although this last step might seem rather restrictive, in our setup we opt for less matches with high quality instead of a higher number of matches with more questionable quality. In summary our algorithm looks as follows:

1. (a) Match the features in I_1 and I_2, using NN matching, optionally with distance check to the second nearest neighbor.
   (b) Match the features in I_2 and I_1, using NN matching, optionally with distance check to the second nearest neighbor.
   (c) Accept only symmetrically matched features.
2. Remove unmatched features in I_1 and merge the remaining features on the basis of the matching in (1.) such that the new cost function between matched features in I_1 and features in I_3 is $\hat{d}(s_{1,l}, s_{3,n}) = \tilde{d}(s_{1,l}, s_{2,m}, s_{3,n})$.
3. (a) Match the features in I_1 and I_3 with the new distance function using NN matching.
   (b) Match the features in I_3 and I_1 with the new distance function using NN matching.
   (c) Accept only symmetrically matched features.
4. Interchange the role of I_1, I_2, I_3 and restart at (1.).
5. Merge the three matchings and return only those matches that are assigned in all three matching directions.

3.4.2 Evaluation of Three-Image Feature Matching

A great number of feature detectors [MTS+05] and feature descriptors [MS05] exist in literature. For a comparison of those we refer the reader to these surveys. The aim of our work is to evaluate the impact of three-image matching and so we chose four widely used detector/ descriptor combinations for our evaluations: SIFT [Low04] and SURF [BETV08] are both scale invariant detectors for blob-like structures and with their natural descriptors also invariant to rotation

and changes in illumination. Additionally we evaluate our matching algorithm on Harris-corners [HS88] and the more recent accelerated corner detector FAST [RD06] and combine both with the normalized cross correlation (NCC) on a 9×9 window. We transform the normalized cross-correlation to a cost function via $d(s_{i,l}, s_{j,m}) = 1 - NCC(f_{i,l}, f_{j,m})$ to obtain a descriptor distance as used in Sect. 3.4.1. Using rather advanced and robust detectors as well as rather low level detectors we want to evaluate our matching scheme independently from the detector used.

For reason of comparison, in our experiments we apply nearest neighbor (NN) matching in all cases [MS05]. Additionally we apply the more advanced NNDR matching that was proposed for SIFT features, using the recommended threshold of 0.8 on the distance ratio [Low04]. NNDR matching could also be applied with the other descriptors, but as the results are very similar we omit them here.

To access the quality of the feature matchings we determine the total number of matched features and the percentage of matches outside a 5 pixel circle around the ground-truth location in different scenes.

We compare our three-image matching strategy (3IM) to two other matching strategies. The first strategy is straightforward two-image matching of the feature descriptors (2IM). For the second strategy we match features between all pairs of images and discard all matches that are not matched consistently on three images (3SF).

As in Sect. 3.3.3 we use the scenes *rubber whale* and *hydrangea* from the Middlebury optical flow dataset [BSL$^+$07], which are monocular sequences with independently moving objects. We also use the scenes *art*, *books*, *dolls*, *laundry*, *moebius* and *reindeer* for evaluation, which are rectified multiple view images of a static scene with known disparity [SS02]. The scenes *waving* and *stonemill* are synthetic, unsynchronized stereo sequences of a moving scene with known ground-truth correspondence fields.

The results in terms of the number of matched features and the percentage of wrong matches among the matched features are given in Tab. 3.4 for SIFT feature

	2IM		3SF		3IM	
	# M	%WM	# M	%WM	# M	%WM
rubber w.	2077	3.85	1628	1.04	1585	0.32
hydr.	1111	16.56	324	5.56	254	2.76
art	1444	53.39	760	26.32	603	11.28
books	1786	15.58	1450	6.62	1373	2.26
dolls	2206	23.75	1677	10.14	1545	4.27
laundry	1112	49.64	645	25.89	550	15.82
moebius	1634	24.24	1208	9.60	1115	5.02
reindeer	943	27.78	725	13.10	664	7.08
waving	4345	11.12	4253	9.92	3995	4.76
stonemill	628	34.71	526	27.00	427	13.11

(a) Using nearest neighbor assignment

	2IM		3SF		3IM	
	# M	%WM	# M	%WM	# M	%WM
rubber w.	1975	0.56	1535	0.39	1510	0.20
hydr.	853	1.52	143	0.70	136	0.74
art	674	10.53	526	4.18	506	2.57
books	1506	2.52	1327	1.06	1315	0.84
dolls	1583	2.21	1407	1.49	1367	1.02
laundry	627	19.94	489	9.82	457	7.66
moebius	1211	4.54	1035	2.9	1011	2.47
reindeer	683	6.88	605	3.97	578	2.77
waving	3804	1.26	3766	1.09	3720	0.70
stonemill	366	2.73	341	1.17	324	0.62

(b) Using nearest neighbor assignment with distance ratio check to the second nearest neighbor

Table 3.4: Using SIFT features and descriptors [Low04], different matching methods are compared: while the two image matching strategy (2IM) matches more features (#M), it has a higher percentage of wrong matches (%WM). Accepting only matches that are consistent on three images (3SF) in a straightforward way reduces the percentage of wrong matches. Our proposed three-image matching strategy (3IM) is able to exclude even more wrong matches.

	2IM		3SF		3IM	
	# M	%WM	# M	%WM	# M	%WM
rubber w.	236	16.53	119	10.08	107	5.61
hydr.	432	20.83	78	16.67	56	8.93
art	616	64.45	229	31.88	177	20.90
books	713	38.85	385	16.10	318	8.81
dolls	809	35.60	503	13.52	434	7.60
laundry	675	68.89	248	38.71	193	28.50
moebius	475	38.95	296	21.62	254	14.96
reindeer	428	43.69	238	21.85	200	14.50
waving	1314	24.20	1246	21.67	1069	12.16
stonemill	251	62.55	169	51.48	114	35.96

(a) Using nearest neighbor assignment

Table 3.5: Using SURF features and descriptors [BETV08], different matching methods are compared: while the two image matching strategy (2IM) matches more features (#M), it has a higher percentage of wrong matches (%WM). Accepting only matches that are consistent in three images (3SF) in a straightforward way reduces the percentage of wrong matches. Our proposed three-image matching strategy (3IM) is able to exclude even more wrong matches.

	2IM		3SF		3IM	
	# M	%WM	# M	%WM	# M	%WM
rubber w.	255	6.67	163	1.84	153	1.31
hydr.	567	22.74	102	12.75	70	8.57
art	474	45.57	263	23.95	220	13.64
books	914	27.02	581	13.25	517	8.70
dolls	812	19.33	581	7.57	528	4.17
laundry	430	55.58	214	28.04	174	17.24
moebius	317	35.65	185	15.14	160	6.88
reindeer	290	33.79	171	18.13	143	11.89
waving	1718	19.97	1644	18.92	1432	9.43
stonemill	763	49.45	668	42.96	452	22.79

(a) Using nearest neighbor assignment

Table 3.6: Using FAST [RD06] features and the color distribution on a 9×9 window as descriptor, different matching methods are compared: while the two image matching strategy (2IM) matches more features (#M), it has a higher percentage of wrong matches (%WM). Accepting only matches that are consistent in three images (3SF) in a straightforward way reduces the percentage of wrong matches. Our proposed three-image matching strategy (3IM) is able to exclude even more wrong matches.

	2IM		3SF		3IM	
	# M	%WM	# M	%WM	# M	%WM
rubber w.	48	0	28	0	25	0
hydr.	176	25.57	31	25.81	20	15.00
art	93	49.46	54	27.78	44	13.64
books	364	21.98	228	12.28	200	9.00
dolls	134	18.66	108	5.56	102	2.94
laundry	158	80.38	47	57.45	32	40.63
moebius	77	20.78	54	7.41	50	4.00
reindeer	49	20.41	40	12.50	37	8.11
waving	196	26.53	176	23.30	156	19.23
stonemill	225	49.78	185	42.16	133	27.82

(a) Using nearest neighbor assignment

Table 3.7: Using Harris corners [HS88] features and the color distribution on a 9×9 window as descriptor, different matching methods are compared: while the two image matching strategy (2IM) matches more features (#M), it has a higher percentage of wrong matches (%WM). Accepting only matches that are consistent in three images (3SF) in a straightforward way reduces the percentage of wrong matches in all but the *Hydrangea* scene. Our proposed three-image matching strategy (3IM) is able to exclude even more wrong matches in all scenes.

with NN matching and NNDR matching, Tab. 3.5 for SURF features, Tab. 3.6 for FAST features and Tab. 3.7 for Harris corners. As a discussion of different features detector/ descriptors and different assignment strategies, NN versus NNDR, is not our intention here, we only discuss the quality of the matchings within one table. The highest number of matches for all descriptors and strategies is achieved by the two-image matching, it is the basis for the two other algorithms. This matching in general contains quite a considerable amount of wrongly assigned matches independently of scene and feature detector, Fig. 3.12. For example SIFT features on the *art* scene, SURF features on the *reindeer* scene, FAST features on the *laundry* scene and Harris corners on the *art* scene each reach a percentage of wrong matches that is over 40%. Considering only matches that the two-image based matching assigns consistently on three images (3SF), the amount of outliers can be reduced in all cases but Harris corners on the *hydrangea* scene where little distinctive small scale structure is provided. As the two-image match is the starting point for the outlier detection, this removal of inconsistent matches reduces the total amount of matches, but in nearly all test-cases it is also able to reduce the percentage of outliers. Evaluating our three-image matching method that considers the third image already at the matching stage, we see that the percentage of outliers is further reduced. In all the examples given above, where the standard two-image matching has a percentage of wrong matches over 40%, our three-image matching is able to reduce the percentage of wrong matches to under 20%. In the case of SURF features on the *reindeer* scene the percentage of outliers can be even reduced from 43.69% to 14.50% removing 158 wrong matches at the cost of 70 correct matches.

Transferring the matching algorithm from sets of equal cardinality to sets with different numbers of features we included some additional assignment steps into our algorithm, Sect. 3.4.1. Tab. 3.8 shows the impact of each of these additional steps for the nearest neighbor (NN) assignment in the case of SIFT features on the *art* and *stonemill* datasets. Other feature detectors/ descriptors and datasets exhibit similar results and are therefore not shown here. Enforcing symmetry of

(a) (b)

Figure 3.12: The two-image based matching approach (a) results in more wrong matches (red circles) and a lower relative amount of correct matches (yellow crosses) than our three-image based matching (b). From top to bottom: scene *art* with SIFT-features, *rubber whale* with SURF-features, *stonemill* with Harris corners, *laundry* with FAST-features, all using nearest neighbor matching.

the matching in step (1.c) and step (3.c) already removes many outlier matches. Intersecting the results of the three-matching from the three different directions, step (5.) removes further outliers and decreases the percentage of wrong matches.

Similar results are obtained when the NNDR assignment is used, Tab. 3.9. Additionally to the impact of the symmetry and the intersection step, i.e., steps (1.c), (3.c) and (5.), we analyze the impact of the distance threshold for the second matching step, i.e., step (3.a)' vs. (3.c)' in Tab. 3.9. While the distance check seems to improve the matching also in the assignment with the modified distance metric $\hat{d}$, the intersection step shows that only very few outliers are removed by the check that would not be removed by the intersection anyway. The removal of valid matches in the *stonemill* datasets leads to an increased percentage of wrong matches so that we do not apply the distance check in our algorithm in the second matching step.

We also test our algorithm on the video sequences *heidelberg*, *skydive*, *market*, *capoeira*, *skateboard* and *jump* from Sect. 3.3.3. The algorithm is performed on the entire images with all feature points found, but for visibility reasons, Fig. 3.13 shows the results only for 100 randomly selected SIFT features: matched features are marked with a white x and connected via a yellow line to the location of the corresponding feature. As features are only matched if they are likely correspondences in three images, the three-matching algorithm obviously decreases the number of matches as compared to the algorithm that matches features based on two images. But our algorithm renounces to match many inconsistent features so that the percentage of outliers is greatly decreased. As we will show in the subsequent sections, this reduction of the relative amount of outliers allows matching based algorithms to start off much better.

Step	Matching	art			stonemill		
		#M	#WM	%WM	#M	#WM	%WM
(1.a)	$I_1 \rightarrow I_2$ (NN)	1444	771	53.39	628	218	34.71
(1.c)	$I_1 \leftrightarrow I_2$ (NN)	1020	360	35.29	500	99	19.80
(3.a)	After $I_1 \leftrightarrow I_3$ (NN): matching $I_1 \rightarrow I_2$ (NN)	1148	559	48.69	581	192	33.05
(3.c)	After $I_1 \leftrightarrow I_3$ (NN): matching $I_1 \leftrightarrow I_2$ (NN)	742	186	25.07	448	73	16.29
(5.)	Intersecting all directions (NN)	603	68	11.28	427	56	13.11

Table 3.8: Contribution of each step of our algorithm to the number of matches (#M), the number of wrong matches (#WM) and the percentage of wrong matches (%WM). Using SIFT-features and nearest neighbor (NN) assignment, the top row shows the result of the two image matching (step (1.a)), which can be improved considering symmetry (step (1.c)). Also, the matching with the modified distance function $\tilde{d}$ can be improved using symmetry (steps (3.a) and (3.c)). Intersection of the results from all matching directions provides the matching with the smallest percentage of wrong matches (step (5.)).

Step	Matching	art			stonemill		
		#M	#WM	%WM	#M	#WM	%WM
(1.a)	$I_1 \rightarrow I_2$ (NNDR)	674	71	10.53	366	10	2.73
(1.c)	$I_1 \leftrightarrow I_2$ (NNDR)	632	43	6.8	350	6	1.71
(3.a)	After $I_1 \leftrightarrow I_3$ (NNDR): matching $I_1 \rightarrow I_2$ (NN)	1007	456	45.28	565	179	31.68
(3.c)	After $I_1 \leftrightarrow I_3$ (NNDR): matching $I_1 \leftrightarrow I_2$ (NN)	712	176	24.72	447	72	16.11
(5.)	Intersecting all directions (NNDR and NN)	506	13	2.57	324	2	0.62
(3.a)'	After $I_1 \leftrightarrow I_3$ (NNDR): matching $I_1 \rightarrow I_2$ (NNDR)	534	23	4.31	346	8	2.31
(3.c)'	After $I_1 \leftrightarrow I_3$ (NNDR): matching $I_1 \leftrightarrow I_2$ (NNDR)	506	13	2.57	325	5	1.54
(5.)'	Intersecting all directions (NNDR only)	495	10	2.02	317	2	0.63

Table 3.9: Applying NNDR assignment for the SIFT features in matching steps (3.a)' and (3.c)' reduces the number of matches (#M) as well as the number of wrong matches (#WM) and the percentage of wrong matches (%WM). But after the intersection of all matching directions (step (5.)'), this turns out to provide only a small advantage in the percentage of the wrong matches (dataset art) or even a disadvantage (dataset stonemill) so we do not apply it in our algorithm.

(a) (b) (c)

Figure 3.13: For the recordings (a) *Heidelberg*, *skydive*, *market*, *capoeira*, *skateboard* and *jump*, two-image and three-image matches are found. For better visibility, here 100 SIFT features are randomly selected and matching features are connected by a yellow line. While (b) two-image matches contain many wrong matches, (c) three-image matching is able to eliminate most the wrong matches.

3.4.3 Discussion

Even in the absence of camera calibration and synchronization, our proposed three-matching algorithm can improve the matching quality by considering three instead of two images simultaneously. However, the increase in the matching quality comes with an increase in computational cost. Straightforward nearest neighbor matching between two images with N features each and d dimensional descriptors has the complexity of $O(dN^2)$. Thus, establishing pairwise feature matches between a set of three images, each having N feature points, has a complexity of $O(3\ dN^2)$. Matching features in forward and backward direction and considering the distances to the third image, our algorithm has the complexity of $O(12\ dN^2)$. The computational burden to the direct approach is increased by a factor of four. In practice, using a naïve MATLAB implementation on a 2.66GHz processor, three image consistent matching of 975 FAST features with 81 dimensional descriptors in I_1, 944 features in I_2 and 860 features in I_3 for the *art* scene requires 736ms. In the same setup, independent two-matching between I_1 and I_2, I_1 and I_3 and I_2 and I_3 lasts together 126ms.

Of course, all validation methods of feature matches come at an increased computational cost. If cameras are synchronized and epipolar geometry between the images can be exploited, a common method to verify matches is to estimate a homography between two images using RANSAC [HZ03]. For example, unidirectional naïve matching and RANSAC verification for three images are on the order of $O(3\ (dN^2 + HN)$, where H is a constant that depends on the number of iterations used in RANSAC [TM97]. The computational cost is, in general, lower than of our three-image matching method, but to be applicable further constraints are imposed on the scene while our algorithm is applicable without additional assumptions. A reduction of the complexity can be achieved by replacing the naïve nearest neighbor search with some faster assignment strategies, e.g., approximate nearest neighbor strategies like best-bin-first search [BL97].

Another drawback of three-image matching is that due to the strict constraints, the number of matched features is reduced - sometimes considerably. In the proposed matching strategy, two-image matches are discarded that might be absolutely correct between two images but merely happen to be undetected or occluded in the third image. This is the price we pay for the increased quality of the matches that are assigned successfully.

The algorithm in Ref. [CS92] is actually designed to approximate solutions to the general k-matching problem for any $k \geq 3$. Accordingly, we can adopt our algorithm to match features in more than three-images to increase the quality of the matching even further. Yet each additional image decreases the ratio of commonly visible and commonly detected feature points. We therefore restrict our evaluation on the minimal example that provides the means to verify the initial two-image match, i.e., we use three images.

In general, if further information on a scene, like epipolar geometry, is available, this should preferably be exploited. The three-image based matching should be rather considered as a last resort to obtain reasonable matchings if further information is not available, e.g. if unsynchronized and uncalibrated cameras are used.

3.5 Three-Image Optical Flow with Feature Matching

Recent optical flow algorithms started to include feature matches into the dense correspondence estimation [BBM09, XJM10, BM10]. As demonstrated in Sect. 2.3 optical flow algorithms need spatial filtering to detect large motions without temporal aliasing. Yet, small, fast moving objects are removed from the image by spatial filtering before their correct motion can be detected. In contrast, feature matching is independent of the absolute position of the features in the image, and can thus deal with large displacements between images - as long as features can be detected and described sufficiently robust. Additionally, most feature descriptors are designed to cope with image noise and changes in illumination. The disadvantage of feature matching is that there are usually only very few features in an image to keep them distinctive. In spite of their relative distinctiveness, feature matches nearly always contain wrong correspondences. As features are matched independently from their relative position in the image, these outliers usually have a large endpoint error. In our approach we unite the advantages of dense correspondence estimation and feature matches. We include feature matches as motion prior into optical flow estimation. Due to the outliers in feature matching, the inclusion of feature based priors harbors the danger of corrupting the quality of the optical flow estimation. We show that the notion of consistency not only provides high quality feature matches and accurate optical flow, but that it also gives the robustness to include feature matches directly into optical flow estimation and thus increases the quality of the optical flow.

3.5.1 Inclusion of Feature Matches into Optical Flow Estimation

In our approach for the integration of feature matching, we include a prior based on the matched features directly into our data-term. Adopting the notation from

Sect. 3.3 with $\mathbf{w}_{i,j}^k$ for the current estimate of the motion field between images I_i and I_j we simply replace the point-wise energy E_q in Eq. (3.10) with

$$E_f = E_q + \delta_f |\mathbf{W}_{i,j} - \mathbf{w}_{i,j}^k - d\mathbf{w}_{i,j}|_2^2 \tag{3.15}$$

where for matches $(f_{i,l}, f_{j,n}, f_{h,m})$ and $[f_{i,l}]$ the nearest integer position to the feature location

$$\mathbf{W}_{i,j} : \Omega \to \Omega, \quad \mathbf{W}_{i,j}(\mathbf{x}) = \begin{cases} f_{j,n} - f_{i,l} & \text{if } \mathbf{x} = [f_{i,l}] \\ 0 & \text{else} \end{cases} \tag{3.16}$$

is a function that describes the matching of the features and for parameters $\mu, c > 0$

$$\delta_f : \Omega \to \mathbb{R}, \quad \delta_f(\mathbf{x}) = \mu \begin{cases} 1 - \frac{\arctan(c\ \tilde{d}(s_{i,l}, s_{j,n}))}{2\pi} & \text{if } \mathbf{x} = [f_{i,l}] \\ 0 & \text{else} \end{cases} \tag{3.17}$$

a weight function. The function δ_f assigns 0 to points that do not have a feature matching assigned. To matched features location it assigns a weight that depends on the matching costs $\tilde{d}(s_{i,l}, s_{j,n})$ for this feature. If the matching costs are high, $\frac{\arctan(c\ \tilde{d}(s_{i,l}, s_{j,n}))}{2\pi}$ is close to 1 while it is close to 0 for very small matching costs.

E_f is still a quadratic function in the update $d\mathbf{w}_{i,j}$, so similar to Sect. 3.3 we can employ a gradient descent scheme. We set $\frac{\partial E_f}{\partial d\mathbf{w}_{1,2,n}} = 0$ for $n \in \{1, 2\}$ and solve the resulting 2×2 linear system for the update $d\mathbf{w}_{1,2}$. If not mentioned otherwise we fix $\mu = 50$ and $c = 5$.

To speed up calculations and assist the determination of large flows, loop-consistent flow estimation is performed on a factor 0.5 image pyramid. Similar to Brox et al. [BBM09] we down-sample the prior $\mathbf{W}_{i,j}$ by considering the 2×2 pixels that are represented by one single pixel in the next coarser level. From the four pixels on the finer level we only pass on to the next coarser level half the motion and the weight of the pixel with the highest weight $\delta_f(x)$. Thus, if no other matches are found in the vicinity, the original match is propagated to the next coarser level or else the match with the smallest cost is used. As matches are only discarded when they interact with other matches, the density

of the matching prior on coarse levels of the image pyramid is much higher than on finer levels.

Having thus established a matching-based prior on all levels of a scale pyramid, we initialize the dense flows on the coarsest level with the prior and perform 10 iterations of the updating scheme with the consistency check from Sect. 3.3 before proceeding to the next finer level. We use the upscaled flow field from the previous level as initialization on the finer level and the prior $\mathbf{W}_{i,j}$ as it is precomputed for the corresponding level and update all flow fields in turn. We proceed in this way until the original resolution is reached.

As features are matched consistently, the prior flow passes the consistency check based on the symmetry and the loop error, Eqs. (3.11) and (3.12). If the prior flow for a pixel is correct, neighboring pixels have corresponding pixels satisfying the brightness constancy. If neighboring pixels cannot find correspondences close to the motion prior, smoothing changes the flow at the points where the motion prior is defined. With increasing difference in the prior motion and the surrounding motion, the prior motion does not satisfy the consistency check anymore. Thus an update due to the motion prior is suppressed. In this way, correct motion priors encourage the flow to converge to the desired motion while wrong matches are suppressed.

3.5.2 Evaluation

To evaluate the impact of three-image-consistent matching on optical flow estimation, we use all the datasets with known ground-truth motion from Sect. 3.3. We measure the average angular error (AAE) and average endpoint error (AEE) [BSL+07] between the computed and the ground-truth displacement fields. For comparison, we also calculate flow fields with a two-image Huber-L^2 approach, Eq. (3.9), incorporating standard two-image feature matching as prior and incorporating three-image feature matching as prior.

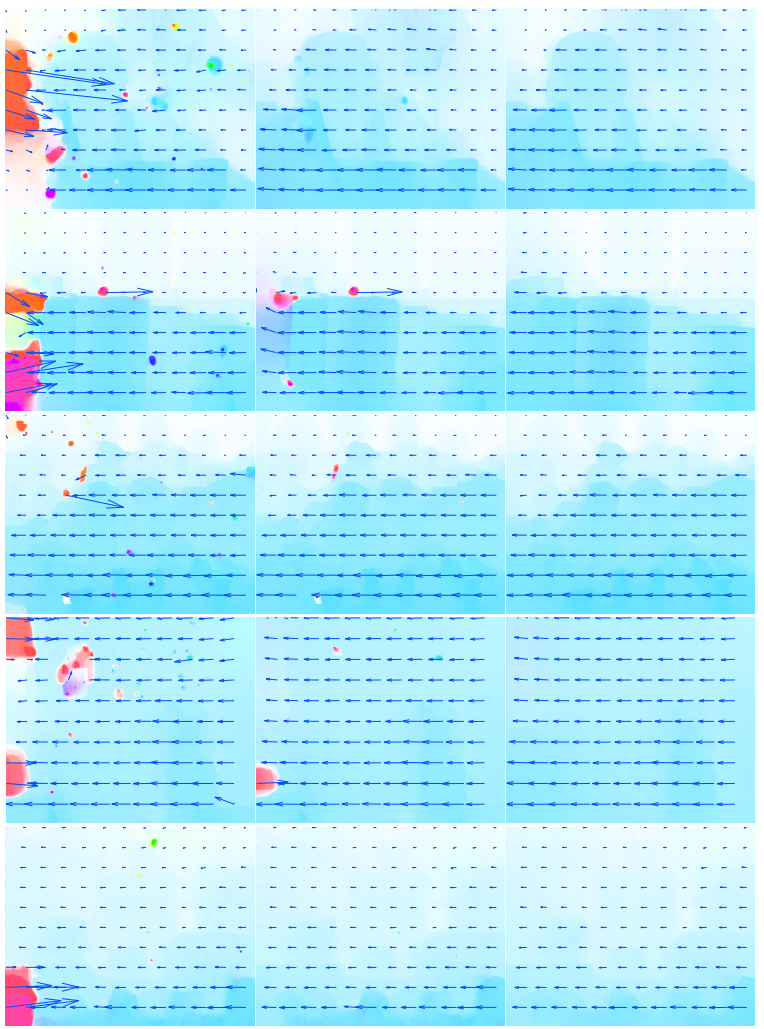
	2IF AAE	2IF & 2IM AAE	2IF & 3IM AAE	3IF AAE	3IF & 3IM AAE
rubber w.	6.01°	6.54°	7.16°	5.64°	5.55°
hydr.	2.78°	2.80°	2.76°	2.58°	2.58°
art	1.07°	11.10°	0.83°	0.88°	0.67°
books	10.63°	10.75°	2.20°	1.34°	0.44°
dolls	0.76°	1.16°	0.49°	0.35°	0.25°
laundry	3.23°	11.18°	2.64°	0.87°	0.73°
moebius	0.69°	6.26°	0.47°	0.55°	0.45°
reindeer	17.99°	2.50°	1.44°	0.93°	0.82°
waving	2.79°	3.54°	2.78°	2.48°	2.26°
stonemill	17.59°	17.48°	16.63°	10.43°	10.00°

(a) The average angular error

	2IF AEE	2IF & 2IM AEE	2IF & 3IM AEE	3IF AEE	3IF & 3IM AEE
rubber w.	0.19 px	0.83 px	1.10 px	0.17 px	0.17 px
hydr.	0.25 px	0.42 px	0.25 px	0.23 px	0.23 px
art	13.54 px	23.20 px	9.48 px	11.22 px	8.96 px
books	15.67 px	14.87 px	5.14 px	4.31 px	4.23 px
dolls	3.72 px	3.41 px	2.21 px	2.62 px	2.00 px
laundry	14.21 px	14.86 px	10.68 px	10.27 px	9.24 px
moebius	4.58 px	9.36 px	3.65 px	4.42 px	4.05 px
reindeer	25.19 px	10.43 px	10.64 px	12.88 px	9.81 px
waving	0.99 px	1.79 px	1.10 px	0.90 px	0.87 px
stonemill	4.65 px	5.63 px	4.94 px	3.53 px	3.49 px

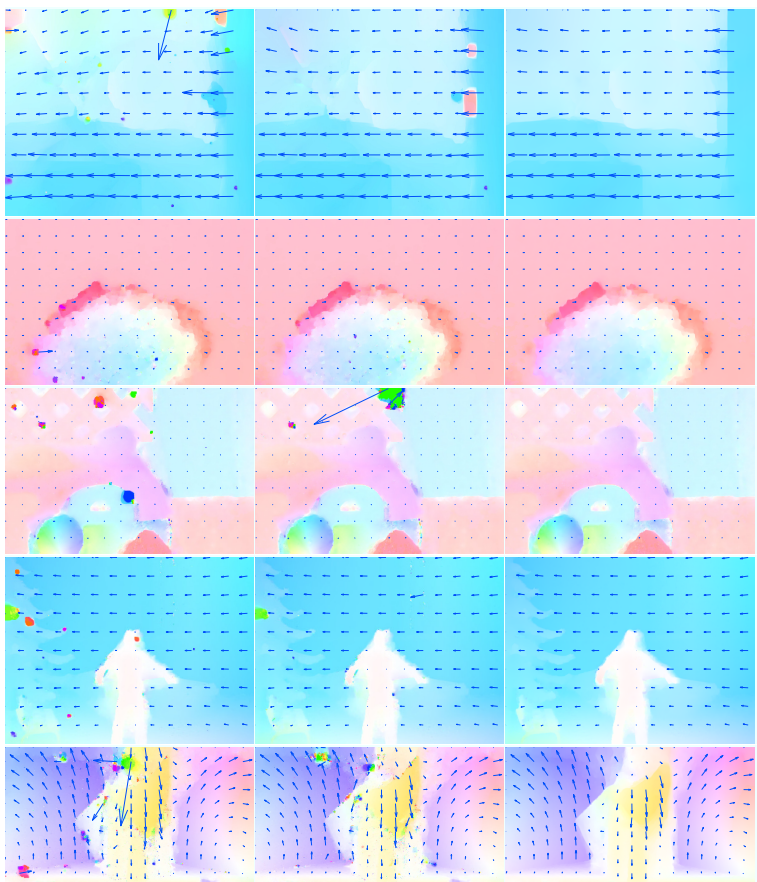
(b) The average endpoint error

Table 3.10: Including 2-image SIFT matching with ratio on the second nearest neighbor as priors into Huber-L^2 flow (2IF& 2IM) significantly increases (a) the average angular (AAE) and (b) the average endpoint error (AEE) in comparison to the basic Huber-L^2 approach (2IF), cf. Tab. 3.1. Including three-image matches into Huber-L^2 flow (2IF& 3IM) improves the quality of the estimated flow in most cases. Under consideration of consistency on a loop of three images, inclusion of 3-image matching priors (3IF & 3IM) decreases the AAE and AEE of the loop-consistent Huber-L^2 approach (3IF), cf. Tab. 3.1.



(a) (b) (c)

Figure 3.14: From top to bottom: results for the datasets *art*, *books*, *dolls*, *laundry* and *moebius* are shown. (a) Using two-image matches in the two-image optical flow algorithm results in large outliers. (b) Three-image matching contains less wrong matches and insertion into two-image optical flow estimation is less noisy. (c) Three-image-consistent flow together with three-image matching yields highest accuracy.



(a) (b) (c)

Figure 3.14: (cont.) From top to bottom: results for the datasets *reindeer*, *hydrangea*, *rubber whale*, *waving* and *stonemill* are shown.

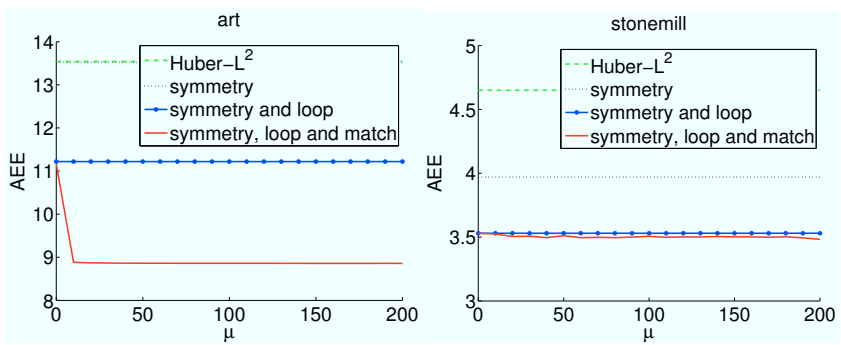
As SIFT features with distance check to the nearest neighbor provide the lowest percentage of outliers, Tab. 3.4(b), and are also applied in related work [XJM10], we use these in our evaluation.

If only two-image matches and forward flow are considered, wrong matches have a strong impact and lead to results with large outlier regions influenced by the wrong matches, Figs. 3.14(a) and 3.15(a), and thus also a high numerical error, Tab. 3.10. In most cases, inclusion of the two-image feature matches decreases the quality of the two-image optical flow. In contrast, using the three-matching strategy from Sect. 3.4 reduces the number of outliers in the matching and is thus able to improve the quality of the two-image flow with feature matches, Tab. 3.10 and Figs. 3.14(b) and 3.15(b) and in most cases also the quality of the two-image optical flow without feature matches. Still, the influence of outlier matches is clearly visible. Further improvement can be obtained by using the three-image feature matches in the three-image-consistent optical flow where the estimated flow additionally has to pass the consistency check, Tab. 3.10 and Fig. 3.14(c). In spite of some outliers in the matching, the correct matches are able to guide the flow so that in all cases the overall average of the angular and the endpoint error of the flow is reduced.

In the evaluation above the parameter $\mu = 50$ for the weight of the matching prior was kept fixed. For the *art* and the *stonemill* datasets we also evaluated the impact of this parameter while keeping all other parameters of the flow fixed. As can be seen in Fig. 3.15 the actual value of the parameter has only a small influence on the resulting flow field as long as it is chosen sufficiently large to have an impact at all.

3.5.3 Discussion

In this section we have included feature matches into our optical flow framework. If only a forward optical flow algorithm and nearest neighbor matching between two images is used, false matches deteriorate the quality of the resulting flow



(a) (b)

Figure 3.15: The choice of the parameter μ which steers the weight of the matching term in the optical flow estimation has little influence on the average endpoint error (AEE): For the datasets (a) *art* and (a) *stonemill* as representative test-cases the AEE of the loop-consistent flow with 3 image feature matching (3IF & 3IM, red, solid) drops below the AEE of the loop-consistent flow without matching prior (3IF, blue, solid with dots), which in turn has a smaller error than the basic Huber-L^2 algorithm (green, dashed) or the flow considering only symmetry (black, dotted).

fields. In contrast, by exploiting the consistency constraint in the flow as well as in the matching, the flow fields increase in quality and the danger of erroneous matches corrupting the flow is banned.

On visual inspection of the flow fields, the difference between loop-consistent flow and loop-consistent flow with feature matching is very small. On the one hand, this is due to the fact that the loop-consistent flow already provides a basically correct estimate of the flow fields. The principal remaining difficulty is the detection of motion of small objects, as these often vanish in the image pyramid before a correct motion can be assigned. On the other hand, small image details often do not contain a feature point that can guide the flow to the correct motion. Of course, the algorithm could include features with a higher density than the proposed features from Sect. 3.4. Unfortunately, denser features are, in general, also less distinctive, producing more ambiguities in the matching and thus do not necessarily lead to a better motion prior for image details.

If features are not assigned automatically by a feature detector, users can generate and match features manually with some mouseclicks. As user interaction guarantees for correct matches, matching costs can be set to zero. The framework can incorporate such additional, reliable features into the optical flow estimation and thus establish dense correspondence fields where correct correspondences cannot be estimated automatically.

3.6 Summary

In this chapter we considered correspondences on sets of three images that are acquired by uncalibrated and unsynchronized cameras. We employ all three images to estimate dense optical flow fields and sparse feature matchings between them more robustly than usual two-image based methods allow. For sparse as well as dense correspondences, we show that wrong correspondences can be considerably reduced and more reliable results can be obtained.

In the dense case we iteratively estimate flow updates based on the usual brightness constancy assumption. Only if the updates are consistent with the other flow fields between the three images these updates are accepted. We observed that the consistency check can suppress spurious flows at occluded points and generally yields optical flow fields with increased accuracy. Additionally we evaluated the consistency check as a confidence measure for optical flow and found that it reliably detects wrong correspondences.

In the sparse case, consistency on three images increases the quality of the feature matches also in those cases where the usual methods for the detection of wrong matches such as epipolar geometry are not applicable.

In the last section of this chapter we combined feature matching and optical flow estimation, exploiting the robustness and the easy handling of arbitrary distances of the former while aiming to improve the accuracy of the latter. Combining two-image feature matches and two-image optical flow considerably decreases the quality of the optical flow. Using instead three-image feature matching improves the accuracy of the two-image flow in most cases. Enforcing consistency on a loop of three images for the matching and the optical flow yields the results with the highest accuracy.

Chapter 4

Conclusion

In this work we introduced two new methods to incorporate additional images into optical flow estimation. The first method focuses on monocular videos and considers an additional motion-blurred image enframed by two sharp short-exposure images. The combination of two different exposure times circumvents temporal aliasing and preserves high frequency detail in motion direction. This way we overcome long-standing limitations of conventional optical flow estimation based on only short-exposure images. In addition to highly accurate motion estimation, our approach allows to estimate occlusion timings. Besides the accuracy of the proposed flow estimation method the availability of occlusion timings is very beneficial in frame interpolation. Even though our estimated flow fields have a remaining error when compared to the ground-truth motion, the estimated occlusion timings allow for frame interpolation that is perceptually even more convincing than interpolation results with ground-truth motion but without occlusion information.

The second proposed method focuses on multi-view video data. We exploit the presence of spatially and temporally neighboring images that have approximately the same view to estimate consistent and accurate dense motion fields and also sparse feature correspondences. With the feedback provided on a loop of three images, many outliers can be successfully excluded and the remaining

loop error can be used as a confidence measure to detect regions where optical flow estimation remains unreliable. Three-image optical flow demonstrates the power of the fill-in effect of the smoothness term. Updates computed based on the brightness constancy assumptions are suppressed if they do not satisfy the loop constraint. In spite of this filtering of the updates, we obtained highly accurate flow fields with only few updating iterations.

Different directions for future work open up based on our contributions and the basic question *How can motion field estimation be improved by new recording setups using additional images?* One issue is the *rolling shutter* that comes about in CMOS sensors of current consumer cameras [WJV+04]. In a rolling shutter camera, lines of pixels are read out one after the other. To ensure equal exposure duration in all lines, exposure time per line starts depending on read-out time. Thus, all lines in an image are acquired with a slight temporal shift. If motion is fast, this leads to distortions in the image. Can the image formation model of the alternate exposure images be adapted to the recording modality of the rolling shutter? Can the combination of distorted short-exposure image, long-exposure image and another distorted short-exposure image be used to estimate the 2D motion and the undistorted images simultaneously? How can occlusion timings be estimated in such a setup?

An issue for future work in multi-view scenarios is the incorporation of user interaction in the flow estimation. Presently, the only influence the user has on flow estimation are the parameter settings such as the weight of the smoothness constraint or the influence of the consistency check. Yet, we have shown that our three-image optical flow estimation provides very similar results quite independently of the parameter values. A more practical interaction should enable the user to directly mark corresponding image regions. With the incorporation of feature matching into our flow algorithm, we already have a framework available that can directly include correspondences indicated by the user. However, user interaction should be applied only to a very small number of images. How

can manually selected correspondences be propagated through multi-view video sequences without introducing any wrong correspondences?

As a last issue for future work the question arises whether there is an optimal number of images to be used in robust feature matching or consistent optical flow. On the one hand, our work has shown that a third image proves to be advantageous. On the other hand, the common field of view and the number of commonly detected features in three images is already smaller than in only two images. Which number of images provides the best trade-off between common information and view differences?

References

[Ado10] Adobe Systems Inc., www.adobe.com, 2010.

[ADPS07] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sanchez. Symmetrical dense optical flow estimation with occlusions detection. *International Journal of Computer Vision*, 75(3):371–385, 2007.

[AN88] J.K. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images-a review. *Proc. of the IEEE*, 76(8):917–935, 1988.

[Ana89] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310, 1989.

[ART10] A. Albarelli, E. Rodolà, and A. Torsello. Robust game-theoretic inlier selection for bundle adjustment. In *Proc. of the International Symposium on 3D Data Processing, Visualization and Transmission*, pages 1–8, Paris, France, May 2010.

[AWS00] L. Alvarez, J. Weickert, and J. Sánchez. Reliable estimation of dense optical flow fields with large displacements. *International Journal of Computer Vision*, 39(1):41–56, 2000.

[AXR09] A. Agrawal, Y. Xu, and R. Raskar. Invertible motion blur in video. In *Proc. SIGGRAPH*, volume 28 of *Transactions on Graphics*, pages 1–8. ACM, Aug. 2009.

[BA91] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 296–302, 1991.

[BA96] M.J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.

[BBM09] T. Brox, C. Bregler, and J. Malik. Large displacement optical flow. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 41–48. IEEE Computer Society, 2009.

[BBPW04] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *Computer Vision-ECCV*, pages 25–36, 2004.

[BBRS07] L. Bar, B. Berkels, M. Rumpf, and G. Sapiro. A variational framework for simultaneous motion estimation and restoration of motion-blurred video. In *Proc. International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[BCS94] H. Bandelt, Y. Crama, and F. Spieksma. Approximation algorithms for multi-dimensional assignment problems with decomposable costs. *Discrete Applied Mathematics*, 49(1-3):25–50, 1994.

[BEN04] M. Ben-Ezra and S.K. Nayar. Motion-based motion deblurring. *Transactions on Pattern Analysis and Machine Intelligence*, 26(6):689, 2004.

[BETV08] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[BFB94] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[BGPS07] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato. SIFT features tracking for video stabilization. In *Proc. of the International Conference on Image Analysis and Processing*, pages 825–830, 2007.

[BJK08] B. Bartczak, D. Jung, and R. Koch. Real-time neighborhood based disparity estimation incorporating temporal evidence. *Pattern Recognition*, pages 153–162, 2008.

[BJNP06] J. Bardsley, S. Jefferies, J. Nagy, and R. Plemmons. Blind iterative restoration of images with spatially-varying blur. *Optics Express*, 14:1767–1782, 2006.

[BKGR09] B. Berkels, C. Kondermann, C. Garbe, and M. Rumpf. Reconstructing optical flow fields by motion inpainting. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, pages 388–400. Springer, 2009.

[BL97] J.S. Beis and D.G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1000–1006. IEEE Computer Society, 1997.

[BM10] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[Bor10] Boris FX, http://www.borisfx.com/red, 2010.

[BSL96] A. Bainbridge-Smith and R.G. Lane. Measuring confidence in optical flow estimation. *Electronics Letters*, 32(10):882–884, may 1996.

[BSL+07] S. Baker, D. Scharstein, JP Lewis, S. Roth, M.J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proc. of the International Conference of Computer Vision (ICCV)*, pages 1–8. IEEE, 2007.

[BSL+09] S. Baker, D. Scharstein, JP Lewis, S. Roth, M.J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. TechReport MSR-TR-2009-179, Microsoft Research, December 2009.

[BTZ96] P. Beardsley, P. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. *Computer Vision—ECCV*, pages 683–695, 1996.

[BW93] R. Bolles and J. Woodfill. Spatiotemporal consistency checking of passive range data. In *Proceedings of International Symposium on Robotics Research*, pages 1–18, Hidden Valley, Pennsylvania, USA, October 1993.

[BW06] A. Bruhn and J. Weickert. A confidence measure for variational optic flow methods. *Geometric Properties for Incomplete data*, pages 283–298, 2006.

[BWSS09] X. Bai, J. Wang, D. Simons, and G. Sapiro. Video snapcut: robust video object cutout using localized classifiers. In *Proc. SIGGRAPH*, volume 28 of *Transactions on Graphics*, pages 1–11. ACM, 2009.

[CGG+06] M. Czerwinski, D.W. Gage, J. Gemmell, C.C. Marshall, M.A. Pérez-Quiñones, M.M. Skeels, and T. Catarci. Digital memories in an era of ubiquitous computing and abundant storage. *Communications of the ACM*, 49(1):44–50, 2006.

[Cha04] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Image Visualization*, 20(1):89–97, 2004.

[Chr00] W.J Christmas. Filtering requirements for gradient-based optical flow measurement. *Transaction on Image Processing*, 9:1817–1820, Oct 2000.

[CNM95] Wei-Ge Chen, N. Nandhakumar, and Worthy N. Martin. Estimating image motion from smear: a sensor system and extensions. In *Proc. of the International Conference on Image Processing (ICIP)*, pages 199–202. IEEE, 1995.

[CNM96] Wei-Ge Chen, N. Nandhakumar, and Worthy N. Martin. Image motion estimation from motion smear-a new computational model. *Transactions on Pattern Analysis and Machine Intelligence*, 18(4), Apr. 1996.

[CS92] Y. Crama and F. C. R. Spieksma. Approximation algorithms for three-dimensional assignment problems with triangle inequalities. *European Journal of Operational Research*, 60(3):273–279, 1992.

[CS05] Tony F. Chan and Jianhong Shen. *Image Processing and Analysis*. siam, Philadelphia, 2005.

[DK02] G.N. DeSouza and A.C. Kak. Vision for mobile robot navigation: A survey. *Transactions on Pattern Analysis and Machine Intelligence*, pages 237–267, 2002.

[DS83] J.E. Dennis and R.B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall Englewood Cliffs, NJ, 1983.

[DW08] S. Dai and Y. Wu. Motion from blur. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE Computer Society, 2008.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[FH04] Pedro Felzenszwalb and Daniel Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2), 2004.

[FMM76] G. E. Forsythe, M. A. Malcolm, and C. B. Moler. *Computer Methods for Mathematical Computations*. Prentice-Hall, 1976.

[FS04] P. Favaro and S. Soatto. A variational approach to scene reconstruction and image segmentation from motion-blur cues. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2004.

[FSH+06] R. Fergus, B. Singh, A. Hertzmann, S.T. Roweis, and W.T. Freeman. Removing camera shake from a single photograph. *ACM Transactions on Graphics*, 2006.

[FTV03] V. Ferrari, T. Tuytelaars, and L. Van Gool. Wide-baseline multiple-view correspondences. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 718–725, Madison, Wisconsin, USA, June 2003.

[GCTC98] A. Giachetti, M. Campani, V. Torre, and C. CRS. The use of optical flow for road navigation. *Transactions on Robotics and Automation*, 14(1):34–48, 1998.

[GJ97] P.R. Giaccone and G.A. Jones. Feed-forward estimation of optical flow. In *Proc. of the International Conference on Image Processing and Its Applications*, volume 1, pages 204 –208. IET, 1997.

[GSC+07] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S.M. Seitz. Multi-view stereo for community photo collections. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 1–8, Rio de Janeiro, Brazil, October 2007. IEEE.

[Had02] Jacques Hadamard. *Sur les problèmes aux dérivées partielles et leur signification physique*. Princeton University Bulletin, 1902.

[HD07] F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 1–7. IEEE, 2007.

[HP96] A.Y.K. Ho and T.C. Pong. Cooperative fusion of stereo and motion. *Pattern Recognition*, 29(1):121–130, 1996.

[HRRS86] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, and W.A. Stahel. *Robust statistics: the approach based on influence functions*, volume 1. Wiley New York, 1986.

[HRT+09] N. Hasler, B. Rosenhahn, T. Thormahlen, M. Wand, J. Gall, and H.P. Seidel. Markerless motion capture with unsynchronized moving cameras. In *Proc. of the Computer Vision and Pattern Recognition (CVPR)*, pages 224–231. IEEE Computer Society, 2009.

[HS81] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[HS88] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. of the Alvey Vision Conference*, volume 15, pages 147–151, Manchester, UK, 1988.

[HZ03] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[IP91] M. Irani and S. Peleg. Improving resolution by image registration. *CVGIP: Graphical Models and Image Processing*, 53(3):231–239, 1991.

[Ira02] M. Irani. Multi-frame correspondence estimation using subspace constraints. *International Journal of Computer Vision*, 48(3):173–194, 2002.

[Jia07] J. Jia. Single image motion deblurring using transparency. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE Computer Society, 2007.

[KH96] D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE Signal Process Magazine*, 13(3):43–64, 1996.

[KKJG07] C. Kondermann, D. Kondermann, B. Jähne, and C. Garbe. An adaptive confidence measure for optical flows based on linear subspace projections. *Pattern Recognition*, pages 132–141, 2007.

[KMG08] C. Kondermann, R. Mester, and C. Garbe. A statistical confidence measure for optical flows. *Computer Vision–ECCV*, pages 290–301, 2008.

[KZK03] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4):359–381, 2003.

[LAG05] S. Lim, J.G. Apostolopoulos, and A.E. Gamal. Optical flow estimation using temporally oversampled video. *Transactions on Image Processing*, 14(8):1074–1087, 2005.

[LALS04] C. Leung, B. Appleton, B.C. Lovell, and C. Sun. An energy minimisation approach to stereo-temporal dense reconstruction. *Pattern Recognition*, 4:72–75, 2004.

[Lev07] A. Levin. Blind motion deblurring using image statistics. *Advances in Neural Information Processing Systems*, 19:841–848, 2007.

[Lip07] L. Lipton. The last great innovation: The stereoscopic cinema. *SMPTE Motion Imaging Journal*, 116(11-12):518–523, 2007.

[LK81] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of the International Joint Conference on Artificial Intelligence*, pages 674–679, Vancouver, Canada, 1981.

[LLB+10] C. Lipski, C. Linz, K. Berger, A. Sellent, and M. Magnor. Virtual video camera: Image-based viewpoint navigation through space and time. *Computer Graphics Forum*, xx(yy):to appear, 2010.

[LLM10] C. Linz, C. Lipski, and M. Magnor. Multi-image interpolation based on graph-cuts and symmetric optic flow. In *Proc. of the International Workshop on Vision, Modeling and Visualization*, pages 115–122, Siegen, Germany, November 2010. Eurographics, Eurographics.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference of Computer Vision (ICCV)*, page 1150. IEEE Computer Society, 1999.

[Low04] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2(60):91–110, 2004.

[LQ02] M. Lhuillier and L. Quan. Match propagation for image-based modeling and rendering. *Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1140–1146, 2002.

[LS08a] R. Li and S. Sclaroff. Multi-scale 3D scene flow from binocular stereo sequences. *Computer Vision and Image Understanding*, 110(1):75–90, 2008.

[LS08b] S.H. Lim and A. Silverstein. Estimation and removal of motion blur by capturing two images with different exposures, 2008.

[LSC+08] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman. Motion-invariant photography. In *Proc. SIGGRAPH*, volume 27 of *Transactions on Graphics*, pages 71:1–71:9. ACM, August 2008.

[LYT+08] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman. Sift flow: dense correspondence across different scenes. *Computer Vision–ECCV 2008*, pages 28–42, 2008.

[Mar91] Robert J. Marks II. *Introduction to Shannon Sampling and Interpolation Theory*. Springer-Verlag, New York, 1991.

[MB96] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: A synopsis of current problems of methods. *International Journal of Computer Vision*, 19(1):29–55, 1996.

[MB04] C. McCarthy and N. Barnes. Performance of optical flow techniques for indoor navigation with a mobile robot. In *International Conference on Robotics and Automation*, volume 5, pages 5093–5098. IEEE, 2004.

[MBW07] Y. Mileva, A. Bruhn, and J. Weickert. Illumination-robust variational optical flow with photometric invariants. In *Proc. of the*

DAGM Conference on Pattern Recognition, pages 152–162. Springer, 2007.

[MF88] M.W. Matlin and H.J. Foley. *Sensation and perception*. Allyn and Bacon Boston, MA, 1988.

[MHM+09] D. Mahajan, F. Huang, W. Matusik, R. Ramamoorthi, and P. Belhumeur. Moving gradients: a path-based method for plausible image interpolation. In *Proc. SIGGRAPH*, volume 28 of *Transaction on Graphics*, pages 1–11, New York, NY, USA, July 2009. ACM.

[MNCG01] B. McCane, K. Novins, D. Crannitch, and B. Galvin. On benchmarking optical flow. *Computer Vision and Image Understanding*, 84(1):126–143, 2001.

[MOG+06] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.Y. Shum. Full-frame video stabilization with motion inpainting. *Transactions on Pattern Analysis and Machine Intelligence*, pages 1150–1163, 2006.

[MP04] W. Matusik and H. Pfister. 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *ACM Transactions on Graphics (TOG)*, 23(3):814–824, 2004.

[MPI96] E. Memin, P. Perez, and R. Irisa. Robust discontinuity-preserving model for estimating optical flow. In *Proc. of the International Conference on Pattern Recognition (ICPR)*, pages 920–924, 1996.

[MS05] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[MS06] D. Min and K. Sohn. Edge-preserving simultaneous joint motion-disparity estimation. In *Proc. of the International Conference on*

Pattern Recognition (ICPR), volume 2, pages 74–77, Washington, DC, USA, 2006. IEEE Computer Society.

[MSMP08] Benjamin Meyer, Timo Stich, Marcus Magnor, and Marc Pollefeys. Subframe temporal alignment of non-stationary cameras. In *Proc. of the British Machine Vision Conference (BMVC)*, 2008.

[MTS+05] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L.V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1):43–72, 2005.

[MYN07] J. Mooser, S. You, and U. Neumann. Real-time object tracking for augmented reality combining graph cuts and optical flow. In *Proc. International Symposium on Mixed and Augmented Reality*, pages 145–152. IEEE, November 2007.

[NDB09] Y. Niu, A. Dick, and M. Brooks. A new inconsistency measure for linear systems and two applications in motion analysis. In *Proc. of the International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 12–17. IEEE, 2009.

[NE86] H.H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, 1986.

[NYC+99] U. Neumann, S. You, Y. Cho, J. Lee, and J. Park. Natural feature tracking for augmented reality. *Transactions on Multimedia*, 1(1):53–64, 1999.

[ON94] M. Otte and H. Nagel. Optical flow estimation: advances and comparisons. *Computer Vision—ECCV*, pages 49–60, 1994.

[PK03] T.L. Pao and M.D. Kuo. Estimation of the point spread function of a motion-blurred object from autocorrelation. In *Proc. of SPIE*, volume 2501, 2003.

[PS98] C.H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Dover Publications, Mineola, New York, USA, 1998.

[RAKRF08] A. Rav-Acha, P. Kohli, C. Rother, and A. Fitzgibbon. Unwrap mosaics: a new representation for video editing. In *Proc. SIGGRAPH*, Transactions on Graphics, pages 1–11. ACM, 2008.

[RAP00] A. Rav-Acha and S. Peleg. Restoration of multiple images with motion blur in differentdirections. In *Workshop on Applications of Computer Vision*, pages 22–28. IEEE, 2000.

[RAP05] A. Rav-Acha and S. Peleg. Two motion-blurred images are better than one. *Pattern Recognition Letters*, 26(3):311–317, 2005.

[RAT06] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: motion deblurring using fluttered shutter. In *Proc. SIGGRAPH*, volume 25 of *Transaction on Graphics*, pages 795–804. ACM, July 2006.

[RD06] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. *Computer Vision–ECCV*, pages 430–443, 2006.

[Rek96] I. M. Rekleitis. Optical flow recognition from the power spectrum of a single blurred image. In *Proc. of the International Conference on Image Processing (ICIP)*, pages 791–794. IEEE, 1996.

[Ric03] I.E.G. Richardson. *H. 264 and MPEG-4 video compression*. Wiley Online Library, 2003.

[ROF92] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[SEG+09] A. Sellent, M. Eisemann, B. Goldlücke, T. Pock, D. Cremers, and M. Magnor. Variational optical flow from alternate exposure images. In *Proc. Vision, Modeling and Visualization (VMV)*, pages 135–143, November 2009.

[SEG+ar] A. Sellent, M. Eisemann, B. Goldlücke, D. Cremers, and M. Magnor. Motion field estimation from alternate exposure images. *Transactions on Pattern Analysis and Machine Intelligence*, to appear.

[SEM09] A. Sellent, M. Eisemann, and M. Magnor. Motion Field and Occlusion Time Estimation via Alternate Exposure Flow. In *Proc. of the International Conference on Computational Photography (ICCP)*. IEEE, April 2009.

[SEM11] A. Sellent, M. Eisemann, and M. Magnor. Robust feature matching in general multi-image setups. *Journal of WSCG*, pages 1–8, February 2011. to appear.

[SK99] C. Stiller and J. Konrad. Estimating motion in image sequences: A tutorial on modeling and computation of 2d motion. *IEEE Signal Processing Magazine*, 16(4):70–91, 1999.

[SLM10] A. Sellent, C. Linz, and M. Magnor. Consistent optical flow for stereo video. In *Proc. of the International Conference on Image Processing (ICIP)*. IEEE, Sept. 2010.

[SLW+10] T. Stich, C. Linz, C. Wallraven, D. Cunningham, and M. Magnor. Perception-motivated interpolation of image sequences. *Transactions on Applied Perception*, pages 1–28, 2010.

[SM04] V. Scholz and M. Magnor. Cloth motion from optical flow. In *Proc. of Vision, Modeling, and Visualization (VMV)*, pages 117– –123, 2004.

[SPC09] F. Steinbruecker, T. Pock, and D. Cremers. Large displacement optical flow computation without warping. In *Proc. of the International Conference of Computer Vision (ICCV)*, Kyoto, Japan, 2009. IEEE.

[Spi00] F.C.R. Spieksma. Multi index assignment problems: complexity, approximation, applications. *Nonlinear Assignment Problems, Algorithms and Applications*, pages 1–12, 2000.

[SS02] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1):7–42, 2002.

[SS05] K. Shafique and M. Shah. A noniterative greedy algorithm for multiframe point correspondence. *Transactions on Pattern Analysis and Machine Intelligence*, pages 51–65, 2005.

[SSS06] N. Snavely, S.M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. In *Proc. SIGGRAPH*, Transactions on Graphics, pages 835–846, Boston, Massachusetts, July 2006. ACM.

[ST08] P. Sand and S. Teller. Particle video: Long-range motion estimation using point trajectories. *International Journal of Computer Vision*, 80(1):72–91, 2008.

[STV08] C. Strecha, T. Tuytelaars, and L. Van Gool. Dense matching of multiple wide-baseline views. In *Proc. of the International Conference on Computer Vision (ICCV)*, pages 1194–1201. IEEE, 2008.

[SW09] M. Sizintsev and R. P. Wildes. Spatiotemporal stereo via spatiotemporal quadric element (stequel) matching. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 493–500, Miami, Floriada, USA, June 2009. IEEE Computer Society.

[SZ02] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". *Computer Vision—ECCV 2002*, pages 414–431, 2002.

[TA77] A.N. Tikhonov and V.Y. Arsenin. *Solutions of Ill-Posed Problems*. Winston, New York, 1977.

[TDBL08] Y.W. Tai, H. Du, M.S. Brown, and S. Lin. Image/video deblurring using a hybrid camera. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE Computer Society, 2008.

[TK96] Damon L. Tull and Aggelos K. Katsaggelos. Regularized blur-assisted displacement field estimation. In *Proc. of the International Conference on Image Processing (ICIP)*, pages 85–88. IEEE, 1996.

[TM97] P.H.S. Torr and D.W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.

[TV06] M. Tico and M. Vehvilainen. Estimation of motion blur point spread function from differently exposed image frames. In *Proc. of Eusipco*, Florence, Italy, September 2006.

[TVG04] T. Tuytelaars and L. Van Gool. Synchronizing video sequences. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–762 – I–768 Vol.1. IEEE Computer Society, june 2004.

[TZ97] P.H.S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing*, 15(8):591–605, 1997.

[Ull79] S. Ullman. *The interpretation of visual motion.* Massachusetts Institut of Technology Press, 1979.

[VBR+05] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 475–480, 2005.

[VBSK00] S. Vedula, S. Baker, S. Seitz, and T. Kanade. Shape and motion carving in 6d. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 592–598. IEEE Computer Society, 2000.

[VP89] A. Verri and T. Poggio. Motion field and optical flow: Qualitative properties. *Transactions on Pattern Analysis and Machine Intelligence*, pages 490–498, 1989.

[VRB03] C.J. Veenman, MJT Reinders, and E. Backer. Establishing motion correspondence using extended temporal scope. *Artificial Intelligence*, 145(1-2):227–243, 2003.

[WAB03] J. Wills, S. Agarwal, and S. Belongie. What Went Where. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1. IEEE Computer Society, 2003.

[WC07] Jue Wang and Michael F. Cohen. Image and video matting: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(2):97–175, 2007.

[Wim10] P. Wimmer. http://www.3dtv.at/movies/, 2010.

[WJV+04] B. Wilburn, N. Joshi, V. Vaish, M. Levoy, and M. Horowitz. High-speed videography using a dense camera array. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2. IEEE, 2004.

[WJV+05] B. Wilburn, N Joshi, V Vaish, E Talvala, E. Anunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *ACM Transactions on Graphics (TOG)*, 24(3):765 – 776, July 2005.

[WPB+09] A. Wedel, T. Pock, J. Braun, U. Franke, and D. Cremers. Duality tv-l1 flow with fundamental matrix prior. In *Proc. of the International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6. IEEE, 2009.

[WPZ+09] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for tv-l 1 optical flow. *Statistical and Geometrical Approaches to Visual Motion Analysis*, pages 23–45, 2009.

[WRV+08] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. *Computer Vision–ECCV*, pages 739–751, 2008.

[WS01] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245–255, 2001.

[WSY07] H. Wang, M. Sun, and R. Yang. Space-time light field rendering. *Transactions on Visualization and Computer Graphics*, pages 697–710, 2007.

[WTP+09] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-L1 optical flow. In *Proc. of the*

British Machine Vision Conference (BMVC), pages 1–11, London, UK, Sept. 2009.

[XCK06] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision*, 67:233–246, 2006.

[XCS+06] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. *Computer Vision–ECCV*, pages 211–224, 2006.

[XJM10] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, 2010. IEEE Computer Society.

[YC07] J. Yao and W.K. Cham. Robust multi-view feature matching from multiple unordered views. *Pattern Recognition*, 40(11):3081–3099, 2007.

[YJS06] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Computing Surveys*, 38(4):13, 2006.

[YK97] Y. Yitzhaky and NS Kopeika. Identification of blur parameters from motion blurred images. *Graphical Models and Image Processing*, 59(5):310–320, 1997.

[YSQS07] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum. Image deblurring with blurred/noisy image pairs. In *Proc. SIGGRAPH*, volume 26 of *Transaction on Graphics*, pages 1–8. ACM, July 2007.

[Zha94] Z. Zhang. Token tracking in a cluttered scene. *Image and Vision Computing*, 12(2):110–120, 1994.

[ZKP10] C. Zach, M. Klopschitz, and M. Pollefeys. Disambiguating visual relations using loop constraints. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9. IEEE Computer Society, June 2010.

[ZPB07] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L^1 optical flow. In *Pattern Recognition*, volume 4713, pages 214–223, 2007.

[ZT03] C. Zhou and H. Tao. Dynamic depth recovery from unsynchronized video streams. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 351–358, June 2003.

Most optical flow algorithms assume pairs of images that are acquired with an ideal, short exposure time. We present two approaches, that use additional images of a scene to estimate highly accurate, dense correspondence fields. In our first approach we consider video sequences that are acquired with alternating exposure times so that a short-exposure image is followed by a long-exposure image that exhibits motion-blur. With the help of the two enframing short-exposure images, we can decipher not only the motion information encoded in the long-exposure image, but also estimate occlusion timings, which are a basis for artifact-free frame interpolation. In our second approach we consider the data modality of multi-view video sequences, as it commonly occurs, e.g., in stereoscopic video. As several images capture nearly the same data of a scene, this redundancy can be used to establish more robust and consistent correspondence fields than the consideration of two images permits.